UPMAIL Technical Report No. 105
May 2, 1995

Simultaneous Rigid E-Unification is Undecidable

Anatoli Degtyarev*
Andrei Voronkov†

Computing Science Department
Uppsala University
Box 311, S-751 05 Uppsala,
Sweden

email {anatoli,voronkov}@csd.uu.se

*Supported by a grant from the Swedish Institute
†Supported by a TFR grant

Abstract

Simultaneous rigid E-unification has been introduced in the area of theorem proving with equality. It is used in *extension procedures,* like the tableau method or the connection method. Many articles in this area assume the existence of an algorithm for simultaneous rigid E-unification. There were several faulty proofs of the decidability of this problem. In this paper we prove that simultaneous rigid E-unification is undecidable. As a consequence we obtain the undecidability of the $\exists^*$-fragment of intuitionistic logic with equality.

1 Introduction

Simultaneous rigid E-unification plays a crucial role in extending to first order languages with equality automatic proof methods based on sequent calculi, such as semantic tableaux [Fitting 88], the connection method [Bibel 82] (also known as the mating method [Andrews 81]), model elimination [Loveland 68] and a dozen other procedures. The usability of simultaneous rigid E-unification has been explained in [Bibel 87, GaRaSn 87]. Since then, there were a number of publications on simultaneous rigid E-unification itself and its use in theorem proving [GNPS 88, GRS 89, GNPS 90, Petermann 90, Baumgartner 92a, Baumgartner 92b, BaFuPe 92, BecHän 92, GNRS 92, Gallier 92, Petermann 92, Beckert 93, Bibel 93, Goubault 93a, Goubault 93b, Goubault 93c, Goubault 93d, Petermann 93a, Petermann 93b, BecPet 94, Beckert 94a, Beckert 94b, Goubault 94, Petermann 94, De Kogel 95].

Some of these articles were based on the conjecture that simultaneous rigid E-unification is decidable. There were several faulty proofs of the decidability of this problem.

Simultaneous rigid E-unification can be formulated as follows. Given equations $s_i = t_i$ and finite sets of equations E_i, $i \in \{1, \ldots, n\}$, find a substitution σ such that $\vdash \forall((\bigwedge_{e \in E_i} e\sigma) \supset s_i\sigma = t_i\sigma)$.

Results on simultaneous rigid E-unification known so far are the following:

1. Non-simultaneous (i.e. the case $n = 1$) rigid E-unification with E_1 ground is NP-complete [Kozen 81].

2. Non-simultaneous rigid E-unification is NP-complete [GNPS 88, GNPS 90].

3. For non-simultaneous rigid E-unification there exist finite complete sets of unifiers [GNPS 88, GNPS 90].

4. The monadic case (the arity of function symbols is ≤ 1) is PSPACE-hard [Goubault 94].

5. The word equation problem is polynomially reducible to monadic simultaneous rigid E-unification [DeMaVo 95].

6. Simultaneous rigid E-unification in the signature with one unary function symbol and a countable number of constants is decidable [DeMaVo 95]. (The decidability proof for this case uses deep number-theoretic facts.)

7. Variable-bounded semi-unification (whose decidability is an open question) is reducible to simultaneous rigid E-unification [DeMaVo 95].

In this paper we prove that simultaneous rigid E-unification is undecidable. The proof is by reduction of so called monadic semi-unification whose undecidability was proven in [Baaz 93][2]. The method of proof uses a modification of the construction proposed in [DeMaVo 95] to prove reduction of variable-bounded semi-unification.

2 Equations and substitutions

The notions of terms, substitutions, equations and rewrite rules are standard and may be found e.g. in [DerJou 90]. An expression (e.g. a term, an equation or a rewrite rule) is *ground* iff it contains

[2]The word "monadic" in monadic semi-unification has a different meaning, it has nothing to do with unary function or predicate symbols.

no variables. The set of all terms of a signature Σ with variables in the set X is denoted by $T_\Sigma(X)$, the set of all ground terms of a signature Σ by T_Σ. We denote

- Equations by $s = t$ (note that $=$ is used to denote the equality predicate);
- Rewrite rules by $s \to t$;

In order to distinguish identity from the equality predicate, the former will be denoted by $\approx$. The notation $\rightleftharpoons$ stands for "equal by definition". For any expression E, $var(E)$ denotes the set of all variables occurring in E. Notation $\vdash$ stands for provability in first-order logic with equality.

Definition 1 *A* substitution *is any mapping θ from a set of variables X to the set $T_\Sigma(X)$ such that there is only a finite number of $x \in X$ with $\theta(x) \not\approx x$. For any expression E and a substitution θ, $E\theta$ stands for the expression obtained from E by the simultaneous replacement of all occurrences of variables $x \in X$ by $\theta(x)$. The* domain *$dom(\theta)$ of the substitution θ is the set of variables $\{x \in X \mid \theta(x) \not\approx x\}$, and the* range *$ran(\theta)$ of θ is the set of terms $\{\theta(x) \mid x \in dom(\theta)\}$. For a substitution σ, $var(\sigma) \rightleftharpoons dom(\sigma) \bigcup var(ran(\sigma))$.*

The composition *$\sigma\theta$ of substitutions σ and θ is the substitution τ defined by $\tau(x) = \theta(\sigma(x))$.*

By $\{t_1/x_1, \dots, t_n/x_n\}$ we denote the substitution σ defined by $\sigma(x_i) \rightleftharpoons t_i$ for $i \in \{1, \dots, n\}$ and $\sigma(x) \rightleftharpoons x$ for $x \notin \{x_1, \dots, x_n\}$. We do not require $x_i \not\approx t_i$.

Composition of substitutions is associative, so we shall omit parentheses in compositions like $\sigma\theta\tau$.

We shall also use the substitution notation $\{t_1/c_1, \dots, t_n/c_n\}$ where c_i are constants. This will denote the operation of the simultaneous replacement of *all* occurrences of c_i by t_i.

Let a set R of rewrite rules of the form $s \to t$ be fixed. The relation of immediate rewriting in R will also be denoted by $\to$. By $\to^*$ we denote the reflexive and transitive closure of the relation $\to$.

Definition 2 *A* normal form *of a term t w.r.t. a set R of rewrite rules is any term $t\downarrow_R$ such that*

1. *$t \to^* t\downarrow_R$;*
2. *There is no term s such that $t\downarrow_R \to s$.*

In general, a normal form is not unique.

Definition 3 *A set R of rewrite rules is called* noetherian *iff the $\to$ relation defined by R is well-founded. The set R has the* diamond property *iff for all terms s, t_1, t_2 such that $s \to t_1$, $s \to t_2$ and $t_1 \not\approx t_2$ there is a term t such that $t_1 \to t$ and $t_2 \to t$. We call a set R of rewrite rules* perfect *iff*

1. *All rewrite rules in R are ground;*
2. *R has the diamond property;*
3. *R is noetherian.*

We shall use the following obvious lemma:

Lemma 1 *Let R be a perfect set of rewrite rules*

$$s_1 \to t_1$$
$$\dots$$
$$s_n \to t_n$$

and E be the equational theory $s_1 = t_1, \dots, s_n = t_n$. Then

1. *For every term t the term $t\downarrow_R$ exists and is unique;*
2. *$E \vdash s = t$ iff $s\downarrow_R \approx t\downarrow_R$.*

3 Simultaneous rigid E-unification

Definition 4 *A* rigid E-unification problem *is any expression of the form $E \vdash_\forall s = t$, where E is a finite set of equations. By $E \models_\forall s = t$ we mean that the formula $\forall((\bigwedge_{e \in E} e) \supset s = t)$ is provable in first-order logic with equality.*

A substitution σ is a solution *(or a rigid E-unifier) of $E \vdash_\forall s = t$ iff $E\sigma \models_\forall (s = t)\sigma$. A* simultaneous rigid E-unification problem *is a finite set $\{\mathcal{E}_1, \dots, \mathcal{E}_n\}$ of rigid E-unification problems. A substitution is its solution iff it is a* solution *of all $\mathcal{E}_i$, $i \in \{1, \dots, n\}$.*

In the proofs below we shall very often implicitly use the following fact: for ground E, s, t we have $E \models_\forall s = t$ iff $E \vdash s = t$.

We shall introduce one particular kind of a rigid E-unification problem that will be used as a technical tool for some proofs in this paper. For a signature Σ, denote by Σ_n the set of all function symbols of arity n in Σ. For every finite signature Σ with at least one constant and a constant $c \in \Sigma_0$ introduce the following set of equations:

$$Gr(\Sigma, c) \rightleftharpoons \bigcup_{n=0}^{\infty} \{f(a_1, \dots, a_n) = c \mid f \in \Sigma_n \setminus \{c\},\ a_1, \dots, a_n \in \Sigma_0\}$$

Lemma 2 *Consider the following rigid E-unification problem:*

$$Gr(\Sigma, c) \vdash_\forall x = c$$

Then a substitution θ is a solution of this problem iff $x\theta \in T_\Sigma$.

Proof. Note that the set R of rewrite rules

$$\bigcup_{n=0}^{\infty} \{f(a_1, \dots, a_n) \to c \mid f \in \Sigma_n \setminus \{c\}\}$$

is perfect. Thus, $Gr(\Sigma, c) \vdash t = c$ iff $t\downarrow_R \approx c\downarrow_R$. It is easy to see that $c\downarrow_R \approx c$ and the set of all such t is exactly the set of all ground terms of Σ.

□

4 Semi-unification

Semi-unification has been introduced in several areas of logic and computer science [LanMus 78, Henglein 88, KMNS 88, Pudlák 88, KfTiUr 89, Leiß 89], most notably because of its relation to the typability problem for polymorphic recursive definitions [Leiß 90]. After several faulty proofs of the decidability of semi-unification, its undecidability was proven in [KfTiUr 93].

For the rest of this paper we assume (without loss of generality) that Σ is a fixed finite signature with at least one constant.

Definition 5 *A* semi-unification *problem is a finite set of expressions*

$$\begin{array}{l} s_1 \leq t_1 \\ \cdots \\ s_n \leq t_n \end{array} \tag{1}$$

where s_i, t_i are terms in the signature Σ. A solution *of problem (1) is a tuple of substitutions $\sigma, \tau_1, \ldots, \tau_n$ in that signature such that*

$$\begin{array}{l} s_1\sigma\tau_1 \approx t_1\sigma \\ \cdots \\ s_n\sigma\tau_n \approx t_n\sigma \end{array} \tag{2}$$

Such a solution is monadic *iff $\mid dom(\tau_i) \mid \leq 1$, for all $i \in \{1, \ldots, n\}$.*

A monadic semi-unification problem *has the same form as a semi-unification problem (1), its solutions are monadic solutions of (1).*

The following theorem is proven in [Baaz 93]:

Theorem 1 *Monadic semi-unification is undecidable.*

For technical reasons, it is convenient for us to have a uniform presentation of monadic solutions. To this end we assume that $\{w_1, w_2, \ldots\}$ is a countable set of variables different from variables of semi-unification problems. The presentation is based on the following lemma:

Lemma 3 *A monadic semi-unification problem $\mathcal{P}$ of the form (1) has a solution iff there is a function $\varphi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ and a solution $\langle \sigma, \tau_1, \ldots, \tau_n \rangle$ of $\mathcal{P}$ such that*

1. *$dom(\sigma) = var(\mathcal{P})$;*
2. *$var(ran(\sigma)) \subseteq \{w_1, \ldots, w_n\}$;*
3. *$var(ran(\tau_i)) \subseteq \{w_1, \ldots, w_n\}$, for all $i \in \{1, \ldots, n\}$;*
4. *$dom(\tau_i) \subseteq \{w_{\varphi(i)}\}$, for all $i \in \{1, \ldots, n\}$.*

Proof. Let $\langle \sigma', \tau'_1, \ldots, \tau'_n \rangle$ be a monadic solution of $\mathcal{P}$. We shall construct a monadic solution satisfying the lemma conditions. Let $U \rightleftharpoons var(\mathcal{P}, \sigma', \tau'_1, \ldots, \tau'_n)$. Without loss of generality we assume $U \bigcap \{w_1, w_2, \ldots\} = \emptyset$. Let $\{x_1, \ldots, x_m\} \rightleftharpoons \bigcup_i dom(\tau'_i)$. It is clear that $m \leq n$. Let c be any constant in Σ. Define a substitution δ with $dom(\delta) \rightleftharpoons U$ by

$$\begin{array}{l} \delta(x_i) \rightleftharpoons w_i \\ \delta(x) \rightleftharpoons c, \quad \text{if } x \in U \setminus \{x_1, \ldots, x_m\} \end{array}$$

Define the function $\varphi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ by

$$\varphi(j) \rightleftharpoons \begin{cases} i, & \text{if } dom(\tau'_j) = \{x_i\} \\ 1, & \text{if } dom(\tau'_j) = \emptyset \end{cases}$$

Let $\sigma, \tau_1, \ldots, \tau_n$ be the substitutions defined by

$$\sigma(x) \rightleftharpoons \begin{cases} x\sigma'\delta, & \text{if } x \in var(\mathcal{P}) \\ x & \text{otherwise} \end{cases}$$

$$\tau_i \rightleftharpoons \begin{cases} \{x\tau'_i\delta/x\delta\}, & \text{if } dom(\tau_i) = \{x\} \\ \tau'_i & \text{if } dom(\tau'_i) = \emptyset \end{cases}$$

We show that $\sigma, \tau_1, \ldots, \tau_n$ and φ satisfy the conditions of the lemma. Conditions 1–3 on variables are simple. Consider condition 4. Let $dom(\tau'_i) = \{x\}$ (the case $dom(\tau'_i) = \emptyset$ is obvious). By the definition of φ, this implies $x \approx x_{\varphi(i)}$. By the definition of δ, we have $x\delta \approx w_{\varphi(i)}$. Thus, $dom(\tau_i) \subseteq \{w_{\varphi(i)}\}$.

Now, let us verify that $\langle \sigma, \tau_1, \ldots, \tau_n \rangle$ is a solution of $\mathcal{P}$, i.e. $s_i\sigma\tau_i \approx t_i\sigma$, for all $i \in \{1, \ldots, n\}$. Since $var(s_i, t_i) \subseteq dom(\sigma)$, this is equivalent to $s_i\sigma'\delta\tau_i \approx t_i\sigma'\delta$. Since $\langle \sigma', \tau'_1, \ldots, \tau'_n \rangle$ is a solution of $\mathcal{P}$, we have $s_i\sigma'\tau'_i \approx t_i\sigma'$, which implies $s_i\sigma'\tau'_i\delta \approx t_i\sigma'\delta$. Thus, it suffices to show $\delta\tau_i = \tau'_i\delta$.

If $dom(\tau'_i) = \emptyset$, it is obvious. Let $dom(\tau'_i) = \{x\}$. We have to prove $y\delta\tau_i \approx y\tau'_i\delta$, for every variable $y \in U$. It is equivalent to $y\delta\{x\tau'_i\delta/x\delta\} \approx y\{x\tau'_i/x\}\delta$. Consider two cases.

1. $y \approx x$. Then $y\delta\{x\tau'_i\delta/x\delta\} \approx x\delta\{x\tau'_i\delta/x\delta\} \approx x\tau'_i\delta \approx x\{x\tau'_i/x\}\delta \approx y\{x\tau'_i/x\}\delta$.
2. $y \not\approx x$. Then $y\delta\{x\tau'_i\delta/x\delta\} \approx y\delta \approx y\{x\tau'_i/x\}\delta$.

□

The following definition is purely technical:

Definition 6 *Let $\mathcal{P}$ be a semi-unification problem of the form (1), $\varphi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ a function. A solution $\langle \sigma, \tau_1, \ldots, \tau_n \rangle$ of $\mathcal{P}$ is a* φ-solution *iff it satisfies all properties (1–4) of Lemma 3.*

A Φ-unification problem *is a pair $(\mathcal{P}, \varphi)$, where $\mathcal{P}$ is a semi-unification problem of the form (1), $\varphi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ is a function. A* solution *of such a problem is any φ-solution of $\mathcal{P}$.*

Lemma 4 *Φ-unification is undecidable.*

Proof. Immediate from Lemma 3: every monadic semi-unification problem is effectively reducible to a finite number of Φ-unification problems.

□

5 The undecidability of simultaneous rigid E-unification

In this section we reduce Φ-unification to simultaneous rigid E-unification.

Let $(\mathcal{P}, \varphi)$ be a Φ-unification problem. Let $u_1, \ldots, u_k$ be the set of all variables occurring in $\mathcal{P}$. Let x_i^j, y_i, v_i^j, z^j be pairwise different variables and $a_{i,l}^j, b_l$ be pairwise different constants not occurring in Σ, where $i \in \{1, \ldots, k\}, j, l \in \{1, \ldots, n\}$.

Define the rigid E-unification problems X_i^j, Y_i, V_i^j, Z^j, where $i \in \{1, \ldots, k\}, j, l \in \{1, \ldots, n\}$ by

$$X_i^j \rightleftharpoons Gr(\Sigma \bigcup \{a_{i,1}^j, \ldots, a_{i,n}^j\}, a_{i,1}^j) \vdash_\forall x_i^j = a_{i,1}^j$$

$$Y_i \rightleftharpoons Gr(\Sigma \bigcup \{b_1, \ldots, b_n\}, b_1) \vdash_\forall y_i = b_1$$

$$V_i^j \rightleftharpoons Gr(\Sigma \bigcup \{b_1, \ldots, b_n\}, b_1) \vdash_\forall v_i^j = b_1$$

$$Z^j \rightleftharpoons Gr(\Sigma \bigcup \{b_1, \ldots, b_n\}, b_1) \vdash_\forall z^j = b_1$$

Note that Lemma 2 applies to all these problems.

In addition, consider the rigid E-unification problems O_i^j, C_i^j and S^j, where $i \in \{1, \ldots, k\}, j \in \{1, \ldots, n\}$ defined by

$$O_i^j \rightleftharpoons a_{i,1}^j = b_1, \ldots, a_{i,n}^j = b_n \vdash_\forall x_i^j = y_i$$

$$C_i^j \rightleftharpoons a_{i,1}^j = b_1, \ldots, a_{i,\varphi(j)-1}^j = b_{\varphi(j)-1}, a_{i,\varphi(j)}^j = z^j, a_{i,\varphi(j)+1}^j = b_{\varphi(j)+1}, \ldots, a_{i,n}^j = b_n \vdash_\forall x_i^j = v_i^j$$

$$S^j \rightleftharpoons \vdash_\forall s_j\{v_1^j/u_1, \ldots, v_k^j/u_k\} = t_j\{y_1/u_1, \ldots, y_k/u_k\}$$

Define the simultaneous rigid E-unification problem

$$\begin{array}{lll} R_{(\mathcal{P},\varphi)} \rightleftharpoons & \{X_i^j \mid i \in \{1, \ldots, k\}, j \in \{1, \ldots, n\}\} & \bigcup \\ & \{Y_i \mid i \in \{1, \ldots, k\}\} & \bigcup \\ & \{V_i^j \mid i \in \{1, \ldots, k\}, j \in \{1, \ldots, n\}\} & \bigcup \\ & \{Z^j \mid j \in \{1, \ldots, n\}\} & \bigcup \\ & \{O_i^j \mid i \in \{1, \ldots, k\}, j \in \{1, \ldots, n\}\} & \bigcup \\ & \{C_i^j \mid i \in \{1, \ldots, k\}, j \in \{1, \ldots, n\}\} & \bigcup \\ & \{S^j \mid j \in \{1, \ldots, n\}\} & \end{array}$$

In the next two lemmas we shall construct a bijection between sets of solutions of $(\mathcal{P}, \varphi)$ and $R_{(\mathcal{P},\varphi)}$.

Lemma 5 *Let $\langle \sigma, \tau_1, \ldots, \tau_n \rangle$ be a solution of $(\mathcal{P}, \varphi)$. Then the substitution θ defined by*

$$\theta(x_i^j) \rightleftharpoons u_i\sigma\{a_{i,1}^j/w_1, \ldots, a_{i,n}^j/w_n\} \tag{3}$$
$$\theta(y_i) \rightleftharpoons u_i\sigma\{b_1/w_1, \ldots, b_n/w_n\} \tag{4}$$
$$\theta(z^j) \rightleftharpoons w_{\varphi(j)}\tau_j\{b_1/w_1, \ldots, b_n/w_n\} \tag{5}$$
$$\theta(v_i^j) \rightleftharpoons u_i\sigma\tau_j\{b_1/w_1, \ldots, b_n/w_n\} \tag{6}$$

is a solution of $R_{(\mathcal{P},\varphi)}$.

Proof. We shall use the tuple notation for terms and substitutions so that (3)–(6) can be rewritten as

$$x_i^j\theta \approx u_i\sigma\{\bar{a}_i^j/\bar{w}\} \tag{7}$$
$$y_i\theta \approx u_i\sigma\{\bar{b}/\bar{w}\} \tag{8}$$
$$z^j\theta \approx w_{\varphi(j)}\tau_j\{\bar{b}/\bar{w}\} \tag{9}$$
$$v_i^j\theta \approx u_i\sigma\tau_j\{\bar{b}/\bar{w}\} \tag{10}$$

We have to prove that θ is a solution of all $X_i^j, Y_i, V_i^j, Z^j, O_i^j, C_i^j, S^j$.

Consider X_i^j. By Lemma 2 it is sufficient to prove $x_i^j\theta \in T_{\Sigma\cup\{a_{i1}^j,\dots,a_{in}^j\}}$. By (7) it is equivalent to $u_i\sigma\{\bar{a}_i^j/\bar{w}\} \in T_{\Sigma\cup\{a_{i1}^j,\dots,a_{in}^j\}}$. This follows from $dom(\sigma) = \{u_1,\dots,u_k\}$ and $var(ran(\sigma)) \subseteq \{w_1,\dots,w_n\}$ (see Definition 6).

The proof that θ is a solution of Y_i, V_i^j, Z^j is similar.

Consider now O_i^j. We have to prove

$$a_{i,1}^j = b_1, \dots, a_{i,n}^j = b_n \models_\forall x_i^j\theta = y_i\theta$$

In view of (7),(8) is it the same as

$$a_{i,1}^j = b_1, \dots, a_{i,n}^j = b_n \models_\forall u_i\sigma\{\bar{a}_i^j/\bar{w}\} = u_i\sigma\{\bar{b}/\bar{w}\}$$

This is obvious.

Consider C_i^j. We have to prove

$$a_{i,1}^j = b_1, \dots, a_{i,\varphi(j)-1}^j = b_{\varphi(j)-1}, a_{i,\varphi(j)}^j = z^j\theta, a_{i,\varphi(j)+1}^j = b_{\varphi(j)+1}, \dots, a_{i,n}^j = b_n \models_\forall x_i^j\theta = v_i^j\theta$$

In view of (9), (7) and (10) is it the same as

$$\begin{array}{c} a_{i,1}^j = b_1, \dots, a_{i,\varphi(j)-1}^j = b_{\varphi(j)-1}, a_{i,\varphi(j)}^j = w_{\varphi(j)}\tau_j\{\bar{b}/\bar{w}\}, a_{i,\varphi(j)+1}^j = b_{\varphi(j)+1}, \dots, a_{i,n}^j = b_n \\ \models_\forall u_i\sigma\{\bar{a}_i^j/\bar{w}\} = u_i\sigma\tau_j\{\bar{b}/\bar{w}\} \end{array} \tag{11}$$

Note that $\tau_j = \{w_{\varphi(j)}\tau_j/w_{\varphi(j)}\}$. Thus, (11) can be rewritten as

$$\begin{array}{l} a_{i,1}^j = b_1, \dots, a_{i,\varphi(j)-1}^j = b_{\varphi(j)-1}, a_{i,\varphi(j)}^j = w_{\varphi(j)}\tau_j\{\bar{b}/\bar{w}\}, a_{i,\varphi(j)+1}^j = b_{\varphi(j)+1}, \dots, a_{i,n}^j = b_n \\ \quad \models_\forall \quad u_i\sigma\{\bar{a}_i^j/\bar{w}\} = \\ \qquad\qquad u_i\sigma\{b_1/w_1, \dots, b_{\varphi(j)-1}/w_{\varphi(j)-1}, w_{\varphi(j)}\tau_j\{\bar{b}/\bar{w}\}/w_{\varphi(j)}, b_{\varphi(j)+1}/w_{\varphi(j)+1}, \dots, b_k/w_k\} \end{array}$$

This is obvious.

Finally, consider S^j. We have to prove

$$\models_\forall s_j\{v_1^j/u_1, \dots, v_k^j/u_k\}\theta = t_j\{y_1/u_1, \dots, y_k/u_k\}\theta$$

Since $var(s_j, t_j) \subseteq \{u_1, \dots, u_k\}$, this is equivalent to

$$\models_\forall s_j\{v_1^j\theta/u_1, \dots, v_k^j\theta/u_k\} = t_j\{y_1\theta/u_1, \dots, y_k\theta/u_k\}$$

In view of (10),(8) is it the same as

$$\models_\forall s_j\{u_1\sigma\tau_j\{\bar{b}/\bar{w}\}/u_1,\ldots,u_k\sigma\tau_j\{\bar{b}/\bar{w}\}/u_k\} = t_j\{u_1\sigma\{\bar{b}/\bar{w}\}/u_1,\ldots,u_k\sigma\{\bar{b}/\bar{w}\}/u_k\}$$

Using $dom(\sigma) = \{u_1,\ldots,u_k\}$, it can be rewritten as

$$\models_\forall s_j\sigma\tau_j\{\bar{b}/\bar{w}\} = t_j\sigma\{\bar{b}/\bar{w}\}$$

This follows from the fact that $\langle\sigma,\tau_1,\ldots,\tau_n\rangle$ is a solution.

□

Lemma 6 *Let a substitution θ be a solution of $R_{(\mathcal{P},\varphi)}$. Then there is a solution $\langle\sigma,\tau_1,\ldots,\tau_n\rangle$ of $(\mathcal{P},\varphi)$ satisfying (7)–(10).*

Proof. Since θ is a solution of X_i^j, Y_i, V_i^j, Z^j and by Lemma 2 we have

$$\begin{aligned} &x_i^j\theta \in T_{\Sigma\cup\{a_{i,1}^j,\ldots,a_{i,n}^j\}} \\ &y_i\theta, v_i^j\theta, z^j\theta \in T_{\Sigma\cup\{b_1,\ldots,b_n\}} \end{aligned} \tag{12}$$

Denote by p_i the term $y_i\theta\{\bar{w}/\bar{b}\}$, where $i \in \{1,\ldots,k\}$. From (12) it follows that $p_i \in T_\Sigma(\{w_1,\ldots,w_n\})$ and

$$y_i\theta \approx p_i\{\bar{b}/\bar{w}\} \tag{13}$$

Consider the following set R of rewrite rules (see the definition of O_i^j):

$$\begin{aligned} &a_{i,1}^j \to b_1 \\ &\ldots \\ &a_{i,n}^j \to b_n \end{aligned}$$

This set is perfect. By Lemma 1, θ is a solution of O_i^j iff $x_i^j\theta\downarrow_R = y_i\theta\downarrow_R$. By (12) we have $y_i\theta\downarrow_R = y_i\theta$. From this and (13) we get

$$x_i^j\theta \approx p_i\{\bar{a}_i^j/\bar{w}\} \tag{14}$$

Denote by q^j the term $z^j\theta\{\bar{w}/\bar{b}\}$, where $j \in \{1,\ldots,n\}$. From (12) it follows that $q^j \in T_\Sigma(\{w_1,\ldots,w_n\})$ and

$$z^j\theta \approx q^j\{\bar{b}/\bar{w}\} \tag{15}$$

Introduce the following substitutions:

$$\sigma \rightleftharpoons \{p_1/u_1,\ldots,p_k/u_k\} \tag{16}$$

$$\tau_j \rightleftharpoons \{q^j/w_{\varphi(j)}\} \tag{17}$$

We are going to prove that $\langle\sigma,\tau_1,\ldots,\tau_n\rangle$ is a solution of $(\mathcal{P},\varphi)$. Note that all conditions of Lemma 3 on variables of σ,τ_i are satisfied. All we have to prove is that $\langle\sigma,\tau_1,\ldots,\tau_n\rangle$ is a solution of $\mathcal{P}$.

Denote by r_i^j the term $v_i^j\theta\{\bar{w}/\bar{b}\}$, where $i \in \{1, \ldots, k\}, j \in \{1, \ldots, n\}$. From (12) it follows that $r_i^j \in T_\Sigma(\{w_1, \ldots, w_n\})$ and

$$v_i^j\theta \approx r_i^j\{\bar{b}/\bar{w}\} \tag{18}$$

Since θ is a solution of C_i^j, we have

$$a_{i,1}^j = b_1, \ldots, a_{i,\varphi(j)-1}^j = b_{\varphi(j)-1}, a_{i,\varphi(j)}^j = z^j\theta, a_{i,\varphi(j)+1}^j = b_{\varphi(j)+1}, \ldots, a_{i,n}^j = b_n \models_\forall x_i^j\theta = v_i^j\theta$$

By (15), (14) and (18) it is equivalent to

$$\begin{aligned} a_{i,1}^j = b_1, \ldots, a_{i,\varphi(j)-1}^j = b_{\varphi(j)-1}, a_{i,\varphi(j)}^j = q^j\{\bar{b}/\bar{w}\}, a_{i,\varphi(j)+1}^j = b_{\varphi(j)+1}, \ldots, a_{i,n}^j = b_n \\ \models_\forall p_i\{\bar{a}_i^j/\bar{w}\} = r_i^j\{\bar{b}/\bar{w}\} \end{aligned} \tag{19}$$

Note that the set R' of rewrite rules

$$\begin{array}{l} a_{i,1}^j \to b_1 \\ \cdots \\ a_{i,\varphi(j)-1}^j \to b_{\varphi(j)-1} \\ a_{i,\varphi(j)}^j \to q^j\{\bar{b}/\bar{w}\} \\ a_{i,\varphi(j)+1}^j \to b_{\varphi(j)+1} \\ \cdots \\ a_{i,n}^j \to b_n \end{array}$$

is perfect. By Lemma 1 and (19) we have

$$p_i\{\bar{a}_i^j/\bar{w}\}\downarrow_{R'} \approx r_i^j\{\bar{b}/\bar{w}\}\downarrow_{R'}$$

It is equivalent to

$$p_i\{b_1/w_1, \ldots, b_{\varphi(j)-1}/w_{\varphi(j)-1}, q^j\{\bar{b}/\bar{w}\}/w_{\varphi(j)}, b_{\varphi(j)+1}/w_{\varphi(j)+1}, \ldots, b_n/w_n\} \approx r_i^j\{\bar{b}/\bar{w}\}$$

which is the same as

$$p_i\{q^j/w_{\varphi(j)}\}\{\bar{b}/\bar{w}\} \approx r_i^j\{\bar{b}/\bar{w}\}$$

This implies

$$p_i\{q^j/w_{\varphi(j)}\} \approx r_i^j$$

By (17) we obtain

$$p_i\tau_j \approx r_i^j \tag{20}$$

Consider now S^j. Since θ is a solution of S^j, we have

$$s_j\{v_1^j/u_1, \ldots, v_k^j/u_k\}\theta \approx t_j\{y_1/u_1, \ldots, y_k/u_k\}\theta$$

This is equivalent to

$$s_j\{v_1^j\theta/u_1, \ldots, v_k^j\theta/u_k\} \approx t_j\{y_1\theta/u_1, \ldots, y_k\theta/u_k\}$$

By (18) and (13) we obtain

$$s_j\{r_1^j\{\bar{b}/\bar{w}\}/u_1, \ldots, r_k^j\{\bar{b}/\bar{w}\}/u_k\} \approx t_j\{p_1\{\bar{b}/\bar{w}\}/u_1, \ldots, p_k\{\bar{b}/\bar{w}\}/u_k\}$$

which implies

$$s_j\{r_1^j/u_1, \ldots, r_k^j/u_k\}\{\bar{b}/\bar{w}\} \approx t_j\{p_1/u_1, \ldots, p_k/u_k\}\{\bar{b}/\bar{w}\}$$

and hence

$$s_j\{r_1^j/u_1, \ldots, r_k^j/u_k\} \approx t_j\{p_1/u_1, \ldots, p_k/u_k\}$$

By (20) this gives

$$s_j\{p_1\tau_j/u_1, \ldots, p_k\tau_j/u_k\} \approx t_j\{p_1/u_1, \ldots, p_k/u_k\}$$

or, equivalently

$$s_j\{p_1/u_1, \ldots, p_k/u_k\}\tau_j \approx t_j\{p_1/u_1, \ldots, p_k/u_k\}$$

By (16) we have

$$s_j\sigma\tau_j \approx t_j\sigma$$

Thus, $\langle \sigma, \tau_1, \ldots, \tau_n \rangle$ is a solution of $\mathcal{P}$.

Now we have to verify (7)–(10). Equation (7) follows from (14) and (16). Equation (8) follows from (13) and (16). Equation (9) follows from (15) and (17). Equation (10) follows from (18), (16) and (20).

□

Lemmas 5 and 6 yield

Theorem 2 *The Φ-unification problem $(\mathcal{P}, \varphi)$ has a solution iff the rigid E-unification problem $R_{(\mathcal{P},\varphi)}$ has a solution.*

Since $(\mathcal{P}, \varphi)$ was arbitrary and by Lemma 4 we obtain

Theorem 3 *Simultaneous rigid E-unification is undecidable.*

In our construction we used a potentially infinite number of new constants $a_{i,l}^j, b_l$. One can note that the proof can as well be modified so that the signature of the simultaneous rigid E-unification problems $R_{(\mathcal{P},\varphi)}$ used in the reduction will be fixed. The modification is as follows. Let $f, g \notin \Sigma$ be unary function symbols and $a \notin \Sigma$ be a constant. Replace all new constants $a_{i,l}^j, b_l$ used in the proofs by terms of the form $fg^m(a)$, where m is a natural number, so that different constants use different m. The proof remains correct because $fg^m(a)$ and $fg^p(a)$ are not subterms of each other when $m \neq p$.

6 The undecidability of the $\exists^*$-fragment of intuitionistic logic with equality

In [GNPS 90] it was noted that "rigid E-unification and Girard's linear logic [Girard 87] share the same spirit". In this section we show that simultaneous rigid E-unification is representable already in a very "weak" fragment of intuitionistic logic with equality, thus obtaining the undecidability result for this fragment. The decidability problems for some other fragments of intuitionistic logic with and without equality were studied in [Orevkov 65, Mints 67, Lifschitz 67, Orevkov 76].

Theorem 4 *The class of formulas provable in intuitionistic logic with equality of the form*

$$\exists x_1 \ldots \exists x_k \Phi$$

where Φ is a quantifier-free formula built from atomic formulas using $\wedge, \supset$, is undecidable.

Proof. We use Theorem 3 on the undecidability of simultaneous rigid E-unification.

Let $E_i \vdash_\forall s_i = t_i$, where $i \in 1, \ldots, n$ be a simultaneous rigid E-unification problem all whose variables are among $x_1, \ldots, x_k$. Consider the formula

$$\exists x_1 \ldots \exists x_k \bigwedge_{i=1}^{n} ((\bigwedge_{e \in E_i} e) \supset s_i = t_i)$$

By the explicit definability property of intuitionistic predicate calculus with equality, this formula is provable iff there are terms $t_1, \ldots, t_k$ such that the formula

$$\bigwedge_{i=1}^{n} ((\bigwedge_{e \in E_i} e) \supset s_i = t_i)\{t_1/x_1, \ldots, t_k/x_k\}$$

is provable.

Denote the substitution $\{t_1/x_1, \ldots, t_k/x_k\}$ by σ. It is easy to see that this formula is provable in intuitionistic logic iff each of the formulas $(\bigwedge_{e \in E_i} e\sigma) \supset s_i\sigma = t_i\sigma$ is provable in intuitionistic logic, which (for this class of formulas) is the same as their provability in classical logic. Provability of all these formulas in classical logic means that σ is a solution of the original simultaneous rigid E-unification problem.

$\square$

Bibliography

[Andrews 81] P.B. Andrews. Theorem proving via general matings. *Journal of the Association for Computing Machinery*, 28(2):193–214, 1981.

[Baaz 93] M. Baaz. Note on the existence of most general semi-unifiers. In *Arithmetic, Proof Theory and Computation Complexity*, volume 23 of *Oxford Logic Guides*, pages 20–29. Oxford University Press, 1993.

[Baumgartner 92a] Peter Baumgartner. An ordered theory resolution calculus. In A. Voronkov, editor, *Logic Programming and Automated Reasoning (LPAR'92)*, volume 624 of *Lecture Notes in Computer Science*, pages 119–130, 1992.

[Baumgartner 92b] Peter Baumgartner. A model elimination calculus with built-in theories. In H.-J. Ohlbach, editor, *Proceedings of the 16th German Workshop on Artificial Intelligence (GWAI'92)*, volume 671 of *Lecture Notes in Computer Science*, pages 30–42, 1992.

[BaFuPe 92] P. Baumgartner, U. Furbach, and U. Petermann. A unified approach to theory reasoning. Forschungsbericht 15/92, Universität Koblenz, 1992.

[Beckert 93] B. Beckert. A completion-based method for adding equality to free variable semantic tableaux. In D. Basin, R. Hähnle, B. Fronhöfer, J. Posegga, and C. Schwind, editors, *Proc. Workshop on Theorem Proving with Analytic Tableaux and Related Methods*, pages 19–22, 1993.

[Beckert 94a] B. Beckert. Adding equality to semantic tableaux. In K. Broda, M. D'Agostino, R. Goré, R. Jahnson, and S. Reeves, editors, *Proceedings, 3rd Workshop on Theorem Proving with Analytic Tableaux and Related Methods, Abingdon*, pages 29–41, Imperial College, London, TR-94/5, 1994.

[Beckert 94b] B. Beckert. A completion-based method for mixed universal and rigid *E*-unification. In A. Bundy, editor, *Automated Deduction — CADE-12. 12th International Conference on Automated Deduction.*, volume 814 of *Lecture Notes in Artificial Intelligence*, pages 678–692, Nancy, France, June/July 1994.

[BecPet 94] G. Becher and U. Petermann. Rigid unification by completion and rigid paramodulation. In B. Nebel and L. Dreschler-Fischer, editors, *KI-94: Advances in Artificial Intelligence. 18th German Annual Conference on Artificial Intelligence*, volume 861 of *Lecture Notes in Artificial Intelligence*, pages 319–330, Saarbrücken, Germany, September 1994. Springer Verlag.

[BecHän 92] B. Beckert and R. Hähnle. An improved method for adding equality to free variable semantic tableaux. In D. Kapur, editor, *11th International Conference on Automated Deduction (CADE)*, volume 607 of *Lecture Notes in Artificial Intelligence*, pages 678–692, Saratoga Springs, NY, USA, June 1992. Springer Verlag.

[Bibel 82] W. Bibel. *Automated theorem proving.* Vieweg Verlag, 1982.

[Bibel 87] W. Bibel. *Automated theorem proving.* Vieweg Verlag, 1987. 2nd edition.

[Bibel 93] W. Bibel. *Deduction. Automated Logic.* Academic Press, 1993.

[DeMaVo 95] A. Degtyarev, Yu. Matijasevich, and A. Voronkov. Simultaneous rigid E-unification is not as simple. UPMAIL Technical Report 104, Uppsala University, Computing Science Department, April 1995.

[De Kogel 95] E. De Kogel. Rigid E-unification simplified. In *Fourth Workshop on Theorem Proving with Analytic Tableaux and Related Methods*, Lecture Notes in Computer Science, May 1995.

[DerJou 90] 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.

[Fitting 88] M. Fitting. First-order modal tableaux. *Journal of Automated Reasoning*, 4:191–213, 1988.

[Gallier 92] J. Gallier. Unification procedures in automated deduction methods based on matings: a survey. In M. Nivat and A. Podelski, editors, *Tree Automata and Languages*, pages 439–485. Elsevier Science, 1992.

[GNPS 88] J.H. Gallier, P. Narendran, D. Plaisted, and W. Snyder. Rigid E-unification is NP-complete. In *Logic in Computer Science (LICS'88) (Edinburgh, Scotland)*, pages 338–346. IEEE Computer Society Press, July 1988.

[GNPS 90] 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.

[GRS 89] J.H. Gallier, S. Raatz, and W. Snyder. Rigid E-unification and its applications to equational matings. In H. Aït Kaci and M. Nivat, editors, *Resolution of Equations in Algebraic Structures*, volume 1, pages 151–216. Academic Press, 1989.

[GNRS 92] J. Gallier, P. Narendran, S. Raatz, and W. Snyder. Theorem proving using equational matings and rigid E-unification. *Journal of the Association for Computing Machinery*, 39(2):377–429, 1992.

[GaRaSn 87] J.H. Gallier, S. Raatz, and W. Snyder. Theorem proving using rigid E-unification: Equational matings. In *Logic in Computer Science (LICS'87) (Ithaca, N.Y.)*, pages 338–346. IEEE Computer Society Press, 1987.

[Girard 87] J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50(1):1–101, 1987.

[Goubault 93a] J. Goubault. *Démonstration Automatique en Logique Classique: Complexité et Méthodes.* PhD thesis, Laboratoire d'informatique de l'école polytechnique, 1993.

[Goubault 93b] J. Goubault. A rule-based algorithm for rigid E-unification. In Georg Gottlob, Alexander Leitsch, and Daniele Mundici, editors, *Computational Logic and Proof Theory. Proceedings of the Third Kurt Gödel Colloquium, KGC'93*, volume 713 of *Lecture Notes in Computer Science*, pages 202–210, Brno, August 1993.

[Goubault 93c] J. Goubault. Simultaneous rigid E-unification is NEXPTIME-complete. Technical Report 93045, Bull, 1993.

[Goubault 93d] J. Goubault. Simultaneous rigid E-unification is NEXPTIME-complete (correction). Technical Report 93047, Bull, 1993.

[Goubault 94] J. Goubault. Rigid $\vec{E}$-unifiability is DEXPTIME-complete. In *Logic in Computer Science (LICS'94)*. IEEE Computer Society Press, 1994.

[Henglein 88] F. Henglein. Type inference and semi-unification. In *Proceedings of the 1988 ACM Conference on LISP and Functional Programming*, pages 184–197, Snowbird, Utah, USA, July 25–27 1988.

[KMNS 88] D. Kapur, D. Musser, P. Narendran, and J. Stillman. Semi-unification. In *Proceedings of the 8th Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 338 of *Lecture Notes in Computer Science*, pages 435–454, Pune, India, December 21–23 1988. Springer Verlag.

[KfTiUr 89] A.J. Kfoury, J. Tiuryn, and P. Urzyczyn. Computational consequences and partial solutions of a generalized unification problem. In *Fourth IEEE Symposium on Logic in Computer Science*, Asilomar, CA, June 5–8 1989.

[KfTiUr 93] A.J. Kfoury, J. Tiuryn, and P. Urzyczyn. The undecidability of the semi-unification problem. *Information and Computation*, 102:83–101, 1993.

[Kozen 81] D. Kozen. Positive first-order logic is NP-complete. *IBM J. of Research and Development*, 25(4):327–332, 1981.

[LanMus 78] D.S. Lankford and D.R. Musser. A finite termination criterion. Technical report, USC Information Sciences Institute, Marina del Rey, CA, 1978.

[Leiß 89] H. Leiß. Semi-unification and type inference for polymorphic recursion. Technical Report INF2-ASE-5-89, Siemens AG, Munich, May 1989.

[Leiß 90] H. Leiß. Polymorphic recursion and semi-unification. In E. Börger, G. Jäger, H. Kleine Büning, and M.M. Richter, editors, *CSL'89 (Proc. 3rd Workshop on Computer Science Logic)*, volume 440 of *Lecture Notes in Computer Science*, pages 211–224, Kaiserslautern, Germany, 1990. Springer Verlag.

[Lifschitz 67] V. Lifschitz. The decidability problem for some constructive theories of equality (in Russian). *Zapiski Nauchnyh Seminarov LOMI*, 4:78–85, 1967.

[Loveland 68] D.W. Loveland. Mechanical theorem proving by model elimination. *Journal of the Association for Computing Machinery*, 15:236–251, 1968.

[Mints 67] G.E. Mints. Collecting terms in the quantifier rules of the constructive predicate calculus (in Russian). *Zapiski Nauchnyh Seminarov LOMI*, 4:78–85, 1967.

[Orevkov 65] V.P. Orevkov. The undecidability in the constructive predicate calculus of the class of formulas of the form $\neg\neg\forall\exists$ (in Russian). *Soviet Mathematical Doklady*, 163(3):581–583, 1965. The journal is cover-to-cover translated to English.

[Orevkov 76] 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.

[Petermann 90] U. Petermann. Towards a connection procedure with built-in theories. In *JELIA'90*, volume 478 of *Lecture Notes in Computer Science*, pages 444–453, 1990.

[Petermann 92] U. Petermann. How to build in an open theory into connection calculi. *J. of Computer and Artificial Intelligence*, 1(2):105–142, 1992.

[Petermann 93a] U. Petermann. A framework for integrating equality reasoning into the extension procedure. In D. Basin, R. Hähnle, B. Fronhöfer, J. Posegga, and C. Schwind, editors, *Proc. Workshop on Theorem Proving with Analytic Tableaux and Related Methods*, pages 195–207, 1993.

[Petermann 93b] U. Petermann. Completeness of the pool calculus with an open built-in theory. In Georg Gottlob, Alexander Leitsch, and Daniele Mundici, editors, *Computational Logic and Proof Theory. Proceedings of the Third Kurt Gödel Colloquium, KGC'93*, volume 713 of *Lecture Notes in Computer Science*, pages 277–288, Brno, August 1993.

[Petermann 94] U. Petermann. A complete connection calculus with rigid E-unification. In *JELIA'94*, volume 838 of *Lecture Notes in Computer Science*, pages 152–166, 1994.

[Pudlák 88] P. Pudlák. On a unification problem related to Kreisel's conjecture. *Mathematicae Universitatis Carolinae*, 29(3):551–556, 1988.