Simultaneous Rigid E-Unification is Undecidable

Anatoli Degtyarev and Andrei Voronkov*

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

Abstract. Simultaneous rigid E-unification was introduced in 1987 by Gallier, Raatz and Snyder. It is used in the area of automated reasoning with equality in *extension procedures,* like the tableau method or the connection method. There were several faulty proofs of the decidability of this problem. We prove the undecidability of simultaneous rigid E-unification using reduction of Hilbert's tenth problem. As a consequence, we obtain the undecidability of the $\exists^*$-fragment of intuitionistic logic with equality and representability of recursively enumerable sets by solutions of simultaneous rigid E-unification.

1 Introduction

The simultaneous rigid E-unification problem plays a crucial role in automatic proof methods for first-order logic with equality based on sequent calculi, such as semantic tableaux [8], model elimination [33], the method of matings [1] (also known as the connection method [9]), and a dozen other procedures. All these methods are based on the Herbrand theorem and express the idea that the proof-search can be considered as the problem of checking that every path through a matrix of the goal formula is inconsistent (for this reason these methods are sometimes characterized as *matrix methods*). This idea was originally justified by Prawitz [43]. Instead of generating instances of a quantifier-free formula $M(\bar{x})$, he proposed to search for a substitution σ for $\bar{x}_1, \ldots, \bar{x}_n$ such that every path in $(M(\bar{x}_1) \wedge \ldots \wedge M(\bar{x}_n))\sigma$ becomes complementary (or inconsistent). In the case without equality, the search for such a substitution can be carried out using ordinary (simultaneous) unification. Thus, for a matrix $M(\bar{x}_1) \wedge \ldots \wedge M(\bar{x}_n)$, the problem of the existence of an appropriate substitution is decidable and can be performed by any known unification algorithm. The idea of reducing the provability (or, dually, unsatisfiability) problem to checking paths in matrices for inconsistency can also be used for logic with equality [30, 22], but the search for a suitable substitution σ results in simultaneous rigid E-unification.

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

* Anatoli Degtyarev is on leave from Glushkov Institute of Cybernetics, Kiev. He is supported by grants from the Swedish Institute and the Swedish Royal Academy of Sciences. Andrei Voronkov is supported by a TFR grant.

provability in first-order logic with equality). The corresponding instance of the simultaneous rigid E-unification problem is denoted by the system of *rigid equations* $E_i \vdash_\forall s_i = t_i$.

Consider an example. Assume that we want to establish the unsatisfiability of the formula $\forall x h(x) = a \land \forall y([h(a) = y \land y \neq g(h(y))] \lor [h(b) = f(y) \land y \neq g(f(y))])$. We can duplicate the quantifier $\forall x$ creating two copies of the subformula $h(x) = a$. Following [1], we can display the resulting matrix in a two-dimensional format, with disjunctions being displayed horizontally and conjunctions being displayed vertically:

$$\left[\begin{array}{c} h(x_1) = a \\ h(x_2) = a \\ \left[\begin{array}{c} h(a) = y \\ y \neq g(h(y)) \end{array}\right] \bigvee \left[\begin{array}{c} h(b) = f(y) \\ y \neq g(f(y)) \end{array}\right] \end{array}\right]$$

Collecting formulas lying on the two vertical paths in this matrix we obtain the following two rigid equations expressing inconsistency of this set of paths

$$\begin{array}{l} h(x_1) = a, h(x_2) = a, h(a) = y \vdash_\forall y = g(h(y)) \\ h(x_1) = a, h(x_2) = a, h(b) = f(y) \vdash_\forall y = g(f(y)) \end{array}$$

Some solutions of this system of rigid equations are $\{a/x_1, b/x_2, g(h^n(a))/y\}$, for every $n \geq 0$.

Since simultaneous rigid E-unification was introduced by Gallier, Raatz and Snyder [24], there have been a number of publications on simultaneous rigid E-unification itself and its use in theorem proving, for example [22, 25, 23, 3, 7, 21, 10, 4, 5, 29, 41]. 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 (e.g. [22, 23, 29]).

The following results on simultaneous rigid E-unification have been known before our first undecidability proof [14]:

1. Non-simultaneous (i.e. the case $n = 1$) rigid E-unification with E_1 ground (i.e. without variables) is NP-complete (Kozen [31]);
2. Non-simultaneous rigid E-unification is NP-complete (Gallier et.al. [22, 23]);
3. For non-simultaneous rigid E-unification there exist finite complete sets of unifiers (Gallier et.al. [22, 23]);
4. The monadic case (the arity of function symbols is ≤ 1) is PSPACE-hard (Gobault [29]);
5. The word equation problem is polynomially reducible to monadic simultaneous rigid E-unification (Matiyasevich and the authors [11, 12]);
6. Simultaneous rigid E-unification in the signature with one unary function symbol and a countable number of constants is decidable (Matiyasevich and the authors [11, 12]). The decidability proof for this case uses deep number-theoretic facts.

In [14] we proved the undecidability of simultaneous rigid E-unification. That first proof used reduction of so called monadic semi-unification whose undecidability was proven by Baaz [2]. A more intuitive proof appeared in [15] using reduction of second-order unification whose undecidability was proven by Goldfarb [28]. Here we give a most elementary proof using reduction of Hilbert's tenth problem. The technique of this proof essentially combines the techniques used by Goldfarb [28] and us [15].

Our paper to CSL'95 was first submitted in May 1995. This version is written in March 1996. In the meantime, several new papers on simultaneous rigid E-unification appeared. They are considered in Section 5.

2 Simultaneous rigid E-unification

Signature in this paper means a finite set of function symbols. *Ground terms* are terms without occurrences of variables. We assume that Σ is a fixed signature containing constants a and `[]` and a binary function symbol `[_|_]`. An *equation* in a signature Δ is an expression of the form $s = t$, where s, t are terms in the signature Δ. The notation $s \approx t$ means that the terms s and t are identical. Variables will always be denoted by x, y, z, u, v, w, maybe with indices.

We assume the standard notion of *substitutions*. Substitutions will be denoted by $\{t_1/x_1, \ldots, t_n/x_n\}$ and mean the operation of the simultaneous replacement of variables x_i by terms t_i. The application of a substitution θ to any expression E (e.g. a term or a set of equations) is denoted by $E\theta$.

For an equation $s = t$ and a set of equations E we write $E \vdash s = t$ to denote that the formula $\forall((\bigwedge_{e \in E} e) \supset s = t)$ is provable in first-order logic with equality. For such formulas provability can be tested by the congruence closure algorithm [44].

Definition 1. A *rigid equation* is an expression of the form $E \vdash_{\forall} s = t$, where E is a finite set of equations and s, t are terms. Its *solution* is any substitution θ such that $E\theta \vdash s\theta = t\theta$. A substitution θ is a solution of a set S of rigid equations iff θ is a solution of every rigid equation in S. A *system* of rigid equations is a finite set of rigid equations.[2].

The *simultaneous rigid E-unification problem* is the problem of determining whether any system of rigid equations possesses a solution.

Let E be an arbitrary expression. Then $E\{c_1 \leftarrow t_1, \ldots, c_n \leftarrow t_n\}$ will denote the expression obtained from E by the simultaneous replacement of all occurrences of constants c_i by terms t_i.

We shall use the following lemma:

Lemma 2. *Let $c_1, \ldots, c_n$ be pairwise different constants and $s_1, s_2, t_1, \ldots, t_n$ be terms such that c_i does not occur in t_j for all $i, j \in \{1, \ldots, n\}$. Then $c_1 =$*

[2] The term "rigid equation" could be more adequately expressed as "instance of a (non-simultaneous) rigid E-unification problem", but this would be too lengthy.

$t_1, \dots, c_n = t_n \vdash s_1 = s_2$ *iff* $s_1\{c_1 \leftarrow t_1, \dots, c_n \leftarrow t_n\} \approx s_2\{c_1 \leftarrow t_1, \dots, c_n \leftarrow t_n\}$.

We shall use the following notation for some special terms. If $t, t_1, \dots, t_n$ are terms, then $[t_1, \dots, t_n \,|\, t]$ denotes the term $[t_1 \,|\, [t_2 \,|\, \dots [t_n \,|\, t] \dots]]$. In particular, when $n = 0$ then $[t_1, \dots, t_n \,|\, t]$ denotes t. If $t_1, \dots, t_n$ are terms, then $[t_1, \dots, t_n]$ denotes the term $[t_1 \,|\, [t_2 \,|\, \dots [t_n \,|\, []] \dots]]$.

3 Undecidability proof

In this section we reduce Hilbert's tenth problem to simultaneous rigid E-unification.

Let $c \notin \Sigma$ be any constant. A *c-numeral* is any term $[a, \dots, a \,|\, c]$. Such a c-numeral with k occurrences of a will be denoted by $a^k c$. We shall use it to represent the natural number k.

For any constant $c \notin \Sigma$ and variable x denote by $Num(x, c)$ the rigid equation

$$c = [a \,|\, c] \vdash_\forall x = c$$

Lemma 3. *A substitution θ is a solution of $Num(x, c)$ iff $x\theta$ is a c-numeral.*

Proof. Obvious.

For any pairwise different constants $b, c, d \notin \Sigma$ and variable x denote by $Lst(x, b, c, d)$ the rigid equation

$$b = [a \,|\, b], c = [a \,|\, c], d = [[b, c] \,|\, d] \vdash_\forall x = d$$

Lemma 4. *A substitution θ is a solution of $Lst(x, b, c, d)$ iff $x\theta$ has the form $[t_1, \dots, t_n \,|\, d]$ for some $n \geq 0$ and t_i of the form $[r_i, s_i]$, where all r_i are b-numerals and all s_i are c-numerals.*

Proof. Similar to that of Lemma 3. QED

In our undecidability proof we shall use reduction of Hilbert's tenth problem. To represent it, we need to represent addition and multiplication on natural numbers. It is done in the following two lemmas.

Let $b, c, d \notin \Sigma$ be pairwise different constants. Denote by $Add(x, y, z, b, c, d)$ the rigid equation

$$d = c, b = y \vdash_\forall z = x$$

Lemma 5. *Let S be the system of rigid equations*

$$\begin{aligned} \{\, & Num(x, b), \\ & Num(y, c), \\ & Num(z, d), \\ & Add(x, y, z, b, c, d)\} \end{aligned}$$

A substitution θ is a solution of S iff for some natural numbers k, l, $x\theta \approx a^k b$, $y\theta \approx a^l c$ and $z\theta \approx a^{k+l} d$.

Proof. Suppose θ is a solution of S. By Lemma 3, the terms $x\theta$, $y\theta$ and $z\theta$ have the forms $a^k b$, $a^l c$ and $a^m d$, respectively. Since θ is a solution of the rigid equation $Add(x, y, z, b, c, d)$, we have

$$d = c, b = a^l c \vdash a^m d = a^k b$$

Lemma 2 yields $a^m c \approx a^k a^l c$. Thus, $m = k + l$.

The "only if" direction is similar. QED

The representation of multiplication is more complicated. We give a simplified construction used in [28] for second-order unification. Let $e, c_1, c_2, c_3, d_1, d_2, d_3 \notin \Sigma$ be pairwise different constants. Then $Mult_1(u, v, c_1, c_2, c_3, d_1, d_2, d_3)$ denotes the rigid equation

$$c_1 = d_1, c_2 = d_2, c_3 = d_3 \vdash_\forall u = v$$

Denote by $Mult_2(u, v, x, y, z, e, c_1, c_2, c_3, d_1, d_2, d_3)$ the rigid equation

$$e = d_1, c_1 = x, c_2 = [a \,|\, d_2], d_3 = [[z, y] \,|\, c_3] \vdash_\forall v = [[d_1, d_2] \,|\, u]$$

Lemma 6. *Let S be the system of rigid equations*

$$\begin{aligned}\{\, & Num(x, e),\\ & Num(y, d_2),\\ & Num(z, d_1),\\ & Lst(u, c_1, c_2, c_3),\\ & Lst(v, d_1, d_2, d_3),\\ & Mult_1(u, v, c_1, c_2, c_3, d_1, d_2, d_3),\\ & Mult_2(u, v, x, y, z, e, c_1, c_2, c_3, d_1, d_2, d_3)\}\end{aligned}$$

A substitution θ is a solution of S iff for some natural numbers k, l, $x\theta \approx a^k e$, $y\theta \approx a^l d_2$ and $z\theta \approx a^{k \cdot l} d_1$.

Proof. Suppose θ is a solution of S. Since θ is a solution of $Lst(u, c_1, c_2, c_3)$, by Lemma 4, the term $u\theta$ has the form

$$[[a^{p_1} c_1, a^{q_1} c_2], \dots, [a^{p_n} c_1, a^{q_n} c_2] \,|\, c_3]$$

Since θ is a solution of $Lst(v, d_1, d_2, d_3)$ and $Mult_1(u, v, c_1, c_2, c_3, d_1, d_2, d_3)$, by Lemmas 2 and 4, the term $v\theta$ has the form

$$[[a^{p_1} d_1, a^{q_1} d_2], \dots, [a^{p_n} d_1, a^{q_n} d_2] \,|\, d_3]$$

Since θ is a solution of $Num(x, e)$, $Num(y, d_2)$ and $Num(z, d_1)$, by Lemma 3, the terms $x\theta$, $y\theta$ and $z\theta$ have the forms $a^k e$, $a^l d_2$ and $a^m d_1$, respectively.

Since θ is a solution of $Mult_2(u, v, x, y, z, e, c_1, c_2, c_3, d_1, d_2, d_3)$ we have

$$e = d_1, c_1 = a^k e, c_2 = [a \,|\, d_2], d_3 = [[a^m d_1, a^l d_2] \,|\, c_3] \vdash v\theta = [[d_1, d_2] \,|\, u\theta]$$

which is equivalent to

$$e = d_1, c_1 = a^k d_1, c_2 = [a \,|\, d_2], d_3 = [[a^m d_1, a^l d_2] \,|\, c_3] \vdash v\theta = [[d_1, d_2] \,|\, u\theta]$$

By Lemma 2,

$$\begin{aligned}&v\theta\{e \leftarrow d_1, c_1 \leftarrow a^k d_1, c_2 \leftarrow [a \,|\, d_2], d_3 \leftarrow [[a^m d_1, a^l d_2] \,|\, c_3]\} \approx \\ &[[d_1, d_2] \,|\, u\theta]\{e \leftarrow d_1, c_1 \leftarrow a^k d_1, c_2 \leftarrow [a \,|\, d_2], d_3 \leftarrow [[a^m d_1, a^l d_2] \,|\, c_3]\}\end{aligned}$$

Substituting for $v\theta$ and $u\theta$, we obtain

$$\begin{aligned}&[[a^{p_1} d_1, a^{q_1} d_2], \dots, [a^{p_n} d_1, a^{q_n} d_2] \,|\, d_3] \\ &\quad \{e \leftarrow d_1, c_1 \leftarrow a^k d_1, c_2 \leftarrow [a \,|\, d_2], d_3 \leftarrow [[a^m d_1, a^l d_2] \,|\, c_3]\} \approx \\ &[[d_1, d_2], [a^{p_1} c_1, a^{q_1} c_2], \dots, [a^{p_n} c_1, a^{q_n} c_2] \,|\, c_3] \\ &\quad \{e \leftarrow d_1, c_1 \leftarrow a^k d_1, c_2 \leftarrow [a \,|\, d_2], d_3 \leftarrow [[a^m d_1, a^l d_2] \,|\, c_3]\}\end{aligned}$$

Replacing the constants c_1, c_2, d_3 by the corresponding terms, we obtain

$$\begin{aligned}&[[a^{p_1} d_1, a^{q_1} d_2], \dots, [a^{p_n} d_1, a^{q_n} d_2], [a^m d_1, a^l d_2] \,|\, c_3] \approx \\ &[[d_1, d_2], [a^{p_1+k} d_1, a^{q_1+1} d_2], \dots, [a^{p_n+k} d_1, a^{q_n+1} d_2] \,|\, c_3]\end{aligned}$$

This implies $p_1 = 0, q_1 = 0, m = p_n + k, l = q_n + 1$ and $p_{i+1} = p_i + k, q_{i+1} = q_i + 1$, for all $i \in \{1, \dots, n-1\}$. Then for all $j \in \{1, \dots, n\}$, $p_j = (j-1) \cdot k$ and $q_j = j - 1$. This yields $m = n \cdot k$ and $l = n$. Hence, $m = k \cdot l$.

The "only if" direction is similar. QED

Lemma 7. *There is an effective method that reduces Hilbert's tenth problem to simultaneous rigid E-unification.*

Proof. Let $\mathcal{D}$ be any finite set of equations of the forms $x_i = 1$, $x_i \cdot x_j = x_k$ and $x_i + x_j = x_k$, where $i, j, k \in \{1, \dots, n\}$. Its solution is any assignment of natural numbers to x_i that makes all equations in $\mathcal{D}$ true. Let $c_1, \dots, c_n, b_1, b_2, b_3, d \notin \Sigma$ be pairwise different constants. It suffices to effectively construct a system of rigid equations $\mathcal{R}$ that is solvable iff $\mathcal{D}$ is solvable.

We shall construct such a system of rigid equations $\mathcal{R}$ using the variables $x_1, \dots, x_n$ and maybe some extra variables. The system of rigid equations $\mathcal{R}$ will satisfy even stronger conditions on its solutions:

(C1) If θ is a solution of $\mathcal{R}$ then $x_i\theta$ is a c_i-numeral;
(C2) For every solution of $\mathcal{D}$, if this solution assigns numbers p_i to x_i, then there is a solution θ of $\mathcal{R}$ such that $x_i\theta \approx a^{p_i} c_i$, for all $i \in \{1, \dots, n\}$;
(C3) For every solution θ of $\mathcal{R}$, if $x_i\theta \approx a^{p_i} c_i$, then the assignment of numbers p_i to x_i is a solution of $\mathcal{D}$.

The system $\mathcal{R}$ consists of the following rigid equations:

1. Rigid equations $Num(x_i, c_i)$, for all $i \in \{1, \dots, n\}$;

2. Rigid equations $\vdash_\forall x_i = [a\,|\,c_i]$ for every equation $x_i = 1$ in $\mathcal{D}$;
3. For every equation $x_i \cdot x_j = x_k$ the rigid equations

$$\begin{array}{l} Lst(u, b_1, b_2, b_3), \\ Lst(v, c_k, c_j, d), \\ Mult_1(u, v, b_1, b_2, b_3, c_k, c_j, d), \\ Mult_2(u, v, x_i, x_j, x_k, c_i, b_1, b_2, b_3, c_k, c_j, d), \end{array}$$

   where u, v are some unique variables (different for every different equation $x_i \cdot x_j = x_k$).
4. Rigid equations $Add(x_i, x_j, x_k, c_i, c_j, c_k)$ for every equation $x_i + x_j = x_k$ in $\mathcal{D}$;

Conditions (C1)–(C3) can easily be checked using Lemmas 3, 5 and 6. QED

In our construction we used a countable number of new constants c_i, b_j, d. One can note that the proof can be modified as well so that we use only a finite number of new symbols. The modification is as follows. Let f, g be unary function symbols and b be a constant all foreign to Σ. Replace all new constants c_i, b_j, d used in the proofs by different terms of the form $f(g^m(b))$. The proof remains correct because $f(g^k(b))$ and $f(g^l(b))$ are not subterms of each other when $k \neq l$. This shows that Hilbert's tenth problem is effectively reducible to simultaneous rigid E-unification in the signature $\Sigma \cup \{f, g, b\}$. By the undecidability of tenth Hilbert's problem [35] we obtain

Theorem 8. *Simultaneous rigid E-unification is undecidable.*

Due to our reduction of Hilbert's tenth problem, we can prove that all recursively enumerable predicates are representable in simultaneous rigid E-unification in the following sense:

Theorem 9. *Let S be any recursively enumerable set of natural numbers. Then there is a system of rigid equations $\mathcal{R}$ in the signature $\Sigma \cup \{f, g, b, c\}$ with a variable x such that*

1. *For every solution θ of $\mathcal{R}$, $x\theta$ is a c-numeral;*
2. *For every natural number n, $n \in S$ iff there is a solution θ of $\mathcal{R}$ such that $x\theta \approx a^n c$.*

Proof. Straightforward from the proof of Theorem 8. QED

As noted recently in [45], it is enough to have one binary function symbol and one constant to represent any Turing machine.

4 The undecidability and recursive completeness of the $\exists^*$-fragment of intuitionistic logic with equality

In [23] Gallier et.al. note that "rigid E-unification and Girard's linear logic [26] 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 by Orevkov [38, 40], Mints [36] and Lifschitz [32].

Theorem 10. *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 8 on the undecidability of simultaneous rigid E-unification.

Let $E_i \vdash_\forall s_i = t_i,\ \ i \in 1, \ldots, n$ be a system of rigid equations 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 $r_1, \ldots, r_k$ such that the formula

$$\bigwedge_{i=1}^{n} ((\bigwedge_{e \in E_i} e) \supset s_i = t_i)\{r_1/x_1, \ldots, r_k/x_k\}$$

is provable. Denote the substitution $\{r_1/x_1, \ldots, r_k/x_k\}$ by σ. It is easy to see that the above 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 rigid equations $E_i \vdash_\forall s_i = t_i$. QED

The sequent calculus and the notion of a cut-free proof for intuitionistic logic with equality are standard and can be found in e.g. in [39, 34, 18, 47]. By combining the proofs of Theorems 9 and 10, we obtain

Theorem 11. *Let S be any recursively enumerable set of natural numbers. Then there is a closed formula ψ of the form $\exists x \exists \bar{y} \varphi(x, \bar{y})$ such that*

1. Every cut-free proof of ψ has the form

$$\dfrac{\begin{array}{c}\vdots \\ \to \exists \bar{y} \varphi(a^n c, \bar{y})\end{array}}{\to \exists x \exists \bar{y} \varphi(x, \bar{y})}$$

2. For every natural number n, $n \in S$ iff there is a cut-free proof of ψ of the above form.

Proof. Straightforward from the proof of Theorem 8. QED

These results are similar to the description of computable functions in the successor arithmetic given by Goad [27], but we do not need any induction axioms.

5 Conclusion

Since our first proof of the undecidability of simultaneous rigid E-unification, other results have been proved. We briefly discuss these results here.

1. Simultaneous rigid E-unification has been originally proposed as a way of handling equality in the method of matings and related methods. It happened that the attractive features of paramodulation-based equality reasoning, such as ordering and basic restrictions, simplification and deletion techniques, are not easy to combine with goal-oriented methods (like semantic tableaux, model elimination or the connection method). In [13, 17, 16] a method not using simultaneous rigid E-unification is introduced for goal-oriented systems. This method of *equality elimination* is based on an extension of top-down procedures by a bottom-up equation solver using basic superposition. A similar approach is developed by Moser, Lynch and Steinbach [37].
2. Plaisted [42] proved that simultaneous rigid E-unification is undecidable even when the left-hand sides of rigid equations are ground. He uses a technique considerably different from ours to reduce the Post Correspondence Problem to simultaneous rigid E-unification. This construction is improved by Veanes [45], where it is shown how to simulate arbitrary Turing machines by simultaneous rigid E-unification with ground left-hand sides. Veanes notes that one binary function symbol, one constant and two variables are enough for proving the undecidability. His result implies that the $\exists\exists$-fragment of intuitionistic logic with equality and function symbols is undecidable which improves our Theorem 10. Plaisted [42] also discusses some substitutes for simultaneous rigid E-unification.
3. Voda and Komara [46] study the problem of *Herbrand skeletons:* for a given formula $\exists \bar{x} \varphi(\bar{x})$ of classical logic with equality and a positive integer n, whether the formula $\varphi(\bar{x}_1) \vee \ldots \vee \varphi(\bar{x}_n)$ has a valid instantiation. Our undecidability result for simultaneous rigid E-unification immediately implies

the undecidability of the Herbrand skeleton problem in the same way as in the proof of Theorem 10. Voda and Komara state that for each particular n, the corresponding problem is also undecidable (our result proves the undecidability only for $n = 1$). They use a reduction of Hilbert's tenth problem similar to ours, but we did not check the details of their proof.
4. In [47] the second author considers the problem of automated theorem proving in intuitionistic logic with equality. It is shown that any complete (semi-decision) procedure for intuitionistic logic with equality gives a semi-decision procedure for simultaneous rigid E-unifiability, and vice versa.
5. In [18] simultaneous rigid E-unification is applied to show the decidability of some fragments of intuitionistic logic with equality. In particular, it is shown that the prenex fragment of intuitionistic logic with equality and without function symbols is PSPACE-complete.
6. The results of [11, 46, 42] show that it is also unlikely to find generally useful decidable subsets of simultaneous rigid E-unification. An alternative approach to use decidable substitutes for this problem has been recently proposed by Petermann [41] and Beckert [6]. Their proposals are based on the restriction on solutions of simultaneous rigid E-unification to be *minimal* in some appropriate sense similar to that of Gallier et.a. [21]. It is an open problem whether the use of minimal solutions results in a complete procedure for classical logic with equality. Using instead of minimal solutions the *basic solutions* to rigid E-unification, in [19] we show that an incomplete but terminating procedure for rigid E-unification gives a complete procedure for classical logic with equality much in the style of the original Gallier et.al.'s proposal [24, 21].
7. In [12] we show that simultaneous rigid E-unification with k unary function symbols is decidable iff it is decidable for $k = 2$. The decidability for this case is still an open problem.

Proofs of the undecidability of simultaneous rigid E-unification, including the one presented in this paper, are summarized in [45].

Acknowledgments

We thank Yuri Matiyasevich for many helpful discussions and suggestions. Gerard Becher, Hans Leiß, Gennady Makanin, Leszek Pacholski, Uwe Petermann, Klaus Schulz and Jerzy Tiuryn have given us valuable explanations on rigid E-unification and some related algorithmic problems. We also thank the program committee of the Tableaux'95 Workshop for stimulating our interest in simultaneous rigid E-unification.

References

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

2. 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.
3. 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.
4. 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.
5. 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.
6. B. Beckert. Are minimal solutions to simultaneous rigid E-unification problems sufficient to adding equality to semantic tableaux. Privately circulated manuscript, 1995.
7. 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.
8. E.W. Beth. *The Foundations of Mathematics*. North Holland, 1959.
9. W. Bibel. On matrices with connections. *Journal of the Association for Computing Machinery*, 28(4):633–645, 1981.
10. W. Bibel. *Deduction. Automated Logic*. Academic Press, 1993.
11. 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.
12. A. Degtyarev, Yu. Matiyasevich, and A. Voronkov. Simultaneous rigid E-unification and related algorithmic problems. To appear in *LICS'96*, 1996.
13. A. Degtyarev and A. Voronkov. Equality elimination for semantic tableaux. UPMAIL Technical Report 90, Uppsala University, Computing Science Department, December 1994. Submitted to a conference.
14. A. Degtyarev and A. Voronkov. Simultaneous rigid E-unification is undecidable. UPMAIL Technical Report 105, Uppsala University, Computing Science Department, May 1995.
15. A. Degtyarev and A. Voronkov. Reduction of second-order unification to simultaneous rigid E-unification. UPMAIL Technical Report 109, Uppsala University, Computing Science Department, June 1995.
16. A. Degtyarev and A. Voronkov. Equality elimination for the inverse method and extension procedures. In C.S. Mellish, editor, *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, volume 1, pages 342–347, Montréal, August 1995.
17. A. Degtyarev and A. Voronkov. General connections via equality elimination. In M. De Glas and Z. Pawlak, editors, *Second World Conference on the Fundamentals of Artificial Intelligence (WOCFAI-95)*, pages 109–120, Paris, July 1995. Angkor.
18. A. Degtyarev and A. Voronkov. Skolemization and decidability problems for fragments of intuitionistic logic. To appear in *LICS'96*, 1996.

19. A. Degtyarev and A. Voronkov. A complete proof procedure for first-order logic based on an incomplete procedure for rigid E-unification. *Submitted to a conference*, 1996.
20. A. Degtyarev and A. Voronkov. The undecidability of simultaneous rigid E-unification. To appear in *Theoretical Computer Science*, 10 pages, 1996.
21. 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.
22. 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.
23. 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.
24. 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.
25. 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.
26. J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50(1):1–101, 1987.
27. C.A. Goad. Computational uses of the manipulation of formal proofs. Technical Report TR no. STAN-CS-80-819, Stanford Univ. Department of Computer Science, 1980.
28. W. D. Goldfarb. The undecidability of the second-order unification problem. *Theoretical Computer Science*, 13:225–230, 1981.
29. J. Goubault. Rigid $\bar{E}$-unifiability is DEXPTIME-complete. In *Logic in Computer Science (LICS'94)*. IEEE Computer Society Press, 1994.
30. S. Kanger. A simplified proof method for elementary logic. In J. Siekmann and G. Wrightson, editors, *Automation of Reasoning. Classical Papers on Computational Logic*, volume 1, pages 364–371. Springer Verlag, 1983. Originally appeared in 1963.
31. D. Kozen. Positive first-order logic is NP-complete. *IBM J. of Research and Development*, 25(4):327–332, 1981.
32. 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.
33. D.W. Loveland. Mechanical theorem proving by model elimination. *Journal of the Association for Computing Machinery*, 15:236–251, 1968.
34. S.Yu. Maslov. Invertible sequential variant of constructive predicate calculus (in Russian). *Zapiski Nauchnyh Seminarov LOMI*, 4:96–111, 1967. English Translation in: Seminars in Mathematics: Steklov Math. Inst. 4, Consultants Bureau, NY-London, 1969, p.36–42.
35. Y.V. Matiyasevič. Enumerable sets are Diophantine. *Soviet Mathematical Doklady*, 11(2):354–358, 1970.
36. 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.

37. M. Moser, C. Lynch, and J. Steinbach. Model elimination with basic ordered paramodulation. Technical Report AR-95-11, Fakultät für Informatik, Technische Universität München, München, 1995.
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. On nonlengthening applications of equality rules (in Russian). *Zapiski Nauchnyh Seminarov LOMI*, 16:152–156, 1969. English Translation in: Seminars in Mathematics: Steklov Math. Inst. 16, Consultants Bureau, NY-London, 1971, p.77-79.
40. 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.
41. 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.
42. 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.
43. D. Prawitz. An improved proof procedure. In J. Siekmann and G. Wrightson, editors, *Automation of Reasoning. Classical Papers on Computational Logic*, volume 1, pages 162–201. Springer Verlag, 1983. Originally appeared in 1960.
44. R. Shostak. An algorithm for reasoning about equality. *Communications of the ACM*, 21:583–585, July 1978.
45. M. Veanes. Undecidability proofs of simultaneous rigid E-unification. Upmail technical report, Uppsala University, Computing Science Department, 1996. To appear.
46. P.J. Voda and J. Komara. On Herbrand skeletons. Technical report, Institute of Informatics, Comenius University Bratislava, July 1995.
47. A. Voronkov. On proof-search in intuitionistic logic with equality, or back to simultaneous rigid E-Unification. UPMAIL Technical Report 121, Uppsala University, Computing Science Department, January 1996. To appear in *CADE'96*, 15 pages, 1996.