On Herbrand Skeletons

Paul J. Voda and Ján Komara

Institute of Informatics, Faculty of Mathematics and Physics,
Comenius University, Mlynská dolina, 842 15 Bratislava, Slovakia.
Technical Report mff-ii-02-1995
July 20, 1995; revised January 26, 1996
{voda,komara}@fmph.uniba.sk

Abstract. Herbrand's theorem plays an important role both in proof theory and in computer science. Given a Herbrand skeleton, which is basically a number specifying the count of disjunctions of the matrix, we would like to get a computable bound on the size of terms which make the disjunction into a quasitautology. This is an important problem in logic, specifically in the complexity of proofs. In computer science, specifically in automated theorem proving, one hopes for an algorithm which avoids the guesses of existential substitution axioms involved in proving a theorem. Herbrand's theorem forms the very basis of automated theorem proving where for a given number n we would like to have an algorithm which finds the terms in the n disjunctions of matrices solely from the shape of the matrix. The main result of this paper is that both problems have negative solutions.

1 Introduction

By the theorem of Herbrand we have for a quantifier-free ϕ:

$$\models \exists \bar{\boldsymbol{x}}\, \phi(\bar{\boldsymbol{x}}) \text{ iff } \models \phi(\bar{\boldsymbol{a}}_1) \lor \phi(\bar{\boldsymbol{a}}_2) \lor \cdots \lor \phi(\bar{\boldsymbol{a}}_n)$$

for certain n and sequences of terms $\bar{\boldsymbol{a}}_1, \ldots, \bar{\boldsymbol{a}}_n$. The question whether for a given n we can find such terms $\bar{\boldsymbol{a}}_i$ for arbitrary formulas ϕ is the problem of *Herbrand skeletons of size n*. The terms $\bar{\boldsymbol{a}}_i$ *solve* the skeleton.

The problem is very important both in logic where we enquire about bounds on the size of such terms [HP93] and also in automated theorem proving (ATP) where one asks whether there is an algorithm which finds the terms from ϕ without a guess involved in the choice of existential substitution axioms. This use of Herbrand's theorem is actually the very foundation of ATP. This is because the theorem of Herbrand has influenced the oldest ATP proof procedure of *resolution* [Rob65] and Herbrand skeletons directly appear in the modern ATP procedures based on the *connection method* (in the form of the *multiplicity* of formulas) [Bib82] and on semantic tableaux (in the form of *free variables*) [Fit90].

It was already known to Herbrand [Her30] that when the formula ϕ does not contain the identity $\doteq$ then we can both effectively find the bounds and that there is an algorithm for finding the terms (unification). With identities permitted in the formulas ϕ, both problems were open although a recent result by Degtyarev and Voronkov on undecidability of the so called *simultaneous rigid E-unification* [DV95a] can be used to show the undecidability for the case when $n = 1$ (see Par. 3.5 for more details). However, this result does not readily extend to the case $n > 1$.

Inspired by [DV95a] and [KP88] we settle in this paper both problems negatively: for no n there is a computable function in ϕ giving a bound on the size of a solution, nor there is an algorithm with input ϕ for finding a solution. We do it by the reduction of the celebrated result of Matiyasevich on unsolvability of Diophantine equations [Mat70] to the solvability of n-skeletons. In order to emphasize the purely logical character of the problem of Herbrand skeletons we deliberately refrain from using any specialized ATP and/or term-rewriting terminology.

In Sect. 2 we give our notation, Sect. 3 introduces the problem of Herbrand skeletons. Section 4 deals with the undecidability of Herbrand skeletons for $n = 1$, Sect. 5 proves technical lemmas needed for this. Section 6 presents the main result for $n \geq 1$, Sect. 7 proves technical lemmas for this.

2 Notation and Logical Background

2.1 Language of predicate calculus. The *language of first-order predicate calculus with identity* consists of denumerably many variables, function and predicate symbols of all arities (function symbols of arity 0 are constants, predicate symbols of arity 0 are propositional constants). We use $\boldsymbol{f}$, and $\boldsymbol{p}$ as metavariables ranging over function and predicate symbols respectively, $\boldsymbol{x}, \boldsymbol{y}, \ldots$, possibly with subscripts, as metavariables ranging over variables.

Semiterms are either variables or expressions $\boldsymbol{f}(\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n)$ where $\boldsymbol{f}$ is a function symbol of arity n and $\boldsymbol{a}_i$ are previously constructed semiterms. We use $\boldsymbol{a}, \boldsymbol{b}, \ldots$ as metavariables ranging over semiterms, $\boldsymbol{k}$ as metavariables ranging over constant symbols, and we write *constant* semiterms $\boldsymbol{k}()$ as $\boldsymbol{k}$.

Any metavariable, say $\boldsymbol{a}$, written as $\bar{\boldsymbol{a}}$, denotes a possibly empty sequence of objects (terms, constants, variables) denoted by the metavariable. Sequences of variables $\bar{\boldsymbol{x}}$, $\bar{\boldsymbol{y}}$ used in a certain context are always assumed to consist of pairwise distinct variables without having any variables in common.

Semiformulas are constructed from *atomic* semiformulas of identity $\boldsymbol{a} \doteq \boldsymbol{b}$ and predicate applications $\boldsymbol{p}(\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n)$ by *propositional* connectives $\neg$, $\vee$, $\wedge$, $\rightarrow$ and *quantifiers* $\exists$, $\forall$ in the usual way. We use lowercase Greek letters $\phi, \psi, \ldots$ to range over semiformulas. Free and bound variables of semiformulas are defined as usual.

Terms are semiterms without variables and *formulas* are semiformulas without free variables. As usual, a *structure* $\mathcal{M}$ is given by a non-empty domain and an interpretation I of function and predicate symbols. We write $\boldsymbol{a}^I$ for the denotation of the term $\boldsymbol{a}$. For a formula ϕ we write $\mathcal{M} \models \phi$ to assert that ϕ is true in the structure $\mathcal{M}$, $\models \phi$ means that ϕ is valid, i.e. that it holds in all structures.

We assume that the objects of first-order predicate calculus are encoded into natural numbers in some of the usual ways. Thus, for instance, the set of terms is a subset of natural numbers. By having variables, semiterms, and semiformulas as natural numbers the metatheoretic predicates, as for instance, $P(\phi)$ are numeric predicates. We may then say that $P(\phi)$ is a recursively enumerable (r.e.) predicate without having to speak of codes of objects. We are also able to write $\boldsymbol{a} = \boldsymbol{b}$ to assert that both terms are identical (note that atomic identity formulas, which are numbers, are written as $\boldsymbol{a} \doteq \boldsymbol{b}$).

2.2 Quasitautologies. *Quasitautologies* are quantifier-free formulas which are tautological (propositional) consequences of *identity axioms*:

$$\boldsymbol{k} \doteq \boldsymbol{k}$$

$$\boldsymbol{a} \doteq \boldsymbol{b} \rightarrow \boldsymbol{b} \doteq \boldsymbol{a}$$

$$\boldsymbol{a} \doteq \boldsymbol{b} \wedge \boldsymbol{b} \doteq \boldsymbol{c} \rightarrow \boldsymbol{a} \doteq \boldsymbol{c}$$

$$\boldsymbol{a}_1 \doteq \boldsymbol{b}_1 \wedge \cdots \wedge \boldsymbol{a}_n \doteq \boldsymbol{b}_n \rightarrow \boldsymbol{f}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \doteq \boldsymbol{f}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$$

$$\boldsymbol{a}_1 \doteq \boldsymbol{b}_1 \wedge \cdots \wedge \boldsymbol{a}_n \doteq \boldsymbol{b}_n \wedge \boldsymbol{p}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \rightarrow \boldsymbol{p}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)\,.$$

For a quantifier-free formula ϕ we have

$$\models \phi \text{ iff } \phi \text{ is a quasitautology.}$$

The predicate of being a quasitautology is primitive recursive (see, for instance, [BJ80]).

2.3 Language of arithmetic. The first-order *language of* arithmetic L consists of the constant $\mathbf{0}$, unary function symbol S, and two binary function symbols $+$, and '$\cdot$'. The structure

$$\mathcal{N} = \langle N, \mathbf{0}, S, +, \cdot \rangle$$

is over the domain of natural numbers with the standard interpretation of the constant $\mathbf{0}$ as zero, S as the successor function, $+$ as addition, and '$\cdot$' as multiplication.

For every constant $\boldsymbol{k}$ we define *$\boldsymbol{k}$-numerals* as terms of the form $S^m(\boldsymbol{k})$ for some $m \geq 0$. Here $S^0(\boldsymbol{a}) = \boldsymbol{a}$ and $S^{m+1}(\boldsymbol{a}) = S\,S^m(\boldsymbol{a})$.

The class of *diophantine semiformulas* is composed from atomic semiformulas $\boldsymbol{a} + \boldsymbol{b} \doteq \boldsymbol{c}$ and $\boldsymbol{a} \cdot \boldsymbol{b} \doteq \boldsymbol{c}$ by conjunctions. Here the terms $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{c}$ are either variables or $\mathbf{0}$-numerals.

By the theorem of Matiyasevich [Mat70] every recursively enumerable predicate $R(m)$ can be represented by a diophantine semiformula $\psi(\boldsymbol{x}, \bar{\boldsymbol{x}})$ with only the indicated variables free such that for every number m

$$R(m) \text{ iff } \mathcal{N} \models \exists \bar{\boldsymbol{x}}\, \psi(S^m(\mathbf{0}), \bar{\boldsymbol{x}})\,. \tag{a}$$

Although the predicate $\mathcal{N} \models \psi$ restricted to diophantine formulas is primitive recursive, the same predicate restricted to existentially closed diophantine semiformulas is only recursively enumerable.

3 Herbrand Skeletons

Herbrand's theorem is the formal basis for ATP. For our purposes it is convenient to state it in terms of solvability of formulas.

3.1 Solvability of formulas. Among the unlimited supply of constant symbols in the language of predicate calculus we single out the constants $*, *_1, *_2, \ldots$ and call them *unknowns*. We use the metavariables $\boldsymbol{*}, \boldsymbol{*}_1, \ldots$ to range over unknowns. Sequences of unknowns are denoted by $\bar{\boldsymbol{*}}$ with the same conventions as for sequences of variables.

We will indicate by writing $\phi(\bar{\boldsymbol{*}})$ that the unknowns occurring in the formula ϕ are among $\bar{\boldsymbol{*}}$. For a sequence of terms $\bar{\boldsymbol{a}}$ of the same length we will write $\phi(\bar{\boldsymbol{a}})$ for the formula obtained by the simultaneous replacement in ϕ of unknowns by the respective terms in $\bar{\boldsymbol{a}}$.

Terms $\bar{\boldsymbol{a}}$ (without unknowns) are called a *solution* of the formula $\phi(\bar{\boldsymbol{*}})$ if $\models \phi(\bar{\boldsymbol{a}})$. A formula ϕ is *solvable* if it has a solution.

We mention some obvious facts about solvability. If $\phi \wedge \psi$ is solvable then both ϕ and ψ are. If ϕ and ψ are solvable and they do not share unknowns then also $\phi \wedge \psi$ is solvable. If ϕ or ψ is solvable then also $\phi \vee \psi$ is. The converse does not hold even if the unknowns are not shared.

3.2 Herbrand's theorem. A formula of the form $\exists \bar{\boldsymbol{x}}\, \phi(\bar{\boldsymbol{x}})$ with ϕ quantifier-free and without unknowns is called *existential formula*. The semiformula ϕ is its *matrice*.

The first part of the theorem of Herbrand says that for every formula ψ we can find an existential formula $\exists \bar{\boldsymbol{x}}\, \phi(\bar{\boldsymbol{x}})$ such that

$$\models \psi \text{ iff } \models \exists \bar{\boldsymbol{x}}\, \phi(\bar{\boldsymbol{x}})\,.$$

This part can proved by the elimination of universal quantifiers from ψ by means of Skolem functions (see, for instance, [Sho67]).

For an existential formula ψ with the matrix $\phi(\bar{\boldsymbol{x}})$ we call any formula

$$\phi(\bar{\boldsymbol{*}}_1) \vee \phi(\bar{\boldsymbol{*}}_2) \vee \cdots \vee \phi(\bar{\boldsymbol{*}}_n) \tag{a}$$

a *Herbrand skeleton of ψ of size n*.

The second part of Herbrand's theorem says that for an existential formula ψ we have

$$\models \psi \text{ iff } \psi \text{ has a solvable skeleton of some size } n.$$

We note that skeletons of the same size differ only in the names of unknowns. Hence, the solvability of any one of them implies the solvability of all skeletons of the same (and larger) size.

Herbrand skeletons are quantifier-free formulas. Thus the test whether a given sequence of terms is a solution to a given skeleton involves the primitive recursive test whether the formula obtained from the skeleton by the replacement of unknowns by the terms is a quasitautology.

We define the predicate $Sk(n, \psi)$ as

$Sk(n, \psi)$ iff ψ is an existential formula with a solvable skeleton of size n.

Hence, for an existential formula ψ we have $\models \psi$ iff $Sk(n, \psi)$ for some n. For every number n we define the predicate $Sk_n(\psi)$ as

$Sk_n(\psi)$ iff $Sk(n, \psi)$.

3.3 Solvability of Herbrand skeletons. Predicate calculus is semi-decidable, i.e. the predicate $\models \psi$ is recursively enumerable but not recursive. By the first part of Herbrand's theorem neither the restriction of $\models \psi$ to existential formulas is recursive. Hence, the equivalent predicate in ψ:

$Sk(n, \psi)$ for some n

is not recursive although it is recursively enumerable.

The recursive enumerability of this predicate has two 'degrees of freedom' as it involves two guesses: first the number n and then the solution. It was hoped in ATP circles that the second guess was not needed and that there was an algorithm which would find a solution or would determine that a formula is unsolvable. In other words, it was hoped that the predicates $Sk_n(\psi)$ were recursive. The main result of this paper is that this is not the case for any $n \geq 1$.

3.4 Example. Consider an existential formula $\exists x\, (p(a) \vee p(b) \to p(x))$ where a and b are different constants. We can solve its 2-skeleton

$$(p(a) \vee p(b) \to p(*)) \vee (p(a) \vee p(b) \to p(*_1))$$

since

$$\models (p(a) \vee p(b) \to p(a)) \vee (p(a) \vee p(b) \to p(b))\,. \tag{a}$$

On the other hand, the 1-skeletons are not solvable.

3.5 Simultaneous rigid E-unification. *A simultaneous rigid E-unification problem* (*SREU problem* for short) is a problem of finding a solution of a formula $\phi(\boldsymbol{*})$ which is a conjunction of n-formulas of the form

$$\boldsymbol{a}_1 \doteq \boldsymbol{b}_1 \wedge \cdots \wedge \boldsymbol{a}_m \doteq \boldsymbol{b}_m \to \boldsymbol{a} \doteq \boldsymbol{b}\,. \tag{a}$$

We do not exclude $m = 0$ in which case the above formula is just the identity $\boldsymbol{a} \doteq \boldsymbol{b}$. For $n = 1$ we have a *rigid E-unification problem*.

It has been known for long that (non-simultaneous) rigid E-unification is decidable (see [GNPS88],[GNPS90]; for a more elementary proof see [Kog95]) while the decidability of SREU has been an open problem. It was recently settled negatively by Degtyarev and Voronkov [DV95a].

The reader will note that the problem of finding a solution of $\phi(\boldsymbol{*})$ is equivalent to the problem of whether the existential formula $\exists \bar{\boldsymbol{x}}\, \phi(\bar{\boldsymbol{x}})$ has a solvable 1-skeleton. Thus the undecidability of SREU implies the undecidability of $Sk_1(\psi)$.

However, the undecidability of $Sk_n(\psi)$ for any n, which is our main result given in Thm. 6.7, is not a direct consequence.

We will outline in Par. 3.6 a procedure, which is a matter of folklore in ATP circles. The procedure converts a Herbrand skeleton of size n to a finite class of SREU problems such that the skeleton is solvable iff at least one SREU from the class is.

When all formulas of a SREU problem are identities, the problem becomes a *(syntactical) unification problem*. That this is decidable was already known to Herbrand [Her30]. Consequently, when an existential formula does not contain the identity $\doteq$ then for any skeleton of size n the conversion procedure yields a finite number of unification problems (i.e. $m = 0$ in each of the problems). Hence, $Sk_n(\psi)$ restricted to such existential formulas is decidable. This has been also known to Herbrand, see also [Bus95a, Bus95b].

3.6 Converting Herbrand skeletons to SREU problems. In this paragraph we reduce in the following sense the problem of solvability of Herbrand skeletons to a class of SREU problems:

> To every quantifier-free formula $\phi(\bar{*})$ we can primitively recursively find a finite class Γ of SREU problems which are *solution equivalent* in the sense that every solution to $\phi(\bar{*})$ solves at least one problem from Γ and vice versa, every solution of a problem from Γ solves $\phi(\bar{*})$.

Let $\phi(\bar{*})$ be a quantifier-free formula. The transformation consists of three steps.

(i): Transformation to conjunction of clauses. We first convert $\phi(\bar{*})$ into a solution equivalent conjunction of clauses $\phi_1(\bar{*})$. A *clause* is of a form

$$A_1 \wedge \cdots \wedge A_n \to B_1 \vee \cdots \vee B_m \tag{a}$$

for atomic formulas $A_1, \ldots, A_n, B_1, \ldots, B_m$. For $m = 1$ the clause is called a *Horn* clause. We do not exclude the case when $n = 0$ in which case (a) stands for $B_1 \vee \cdots \vee B_m$ or the case when $m = 0$ in which case we put the formula $c \doteq d$ for two *new* distinct constants in the consequent (body) of the clause.

We set $\Gamma_0 = \{\phi_1(\bar{*})\}$. By definition Γ_0 and $\phi(\bar{*})$ are solution equivalent.

(ii): Transformation to Horn clauses. Assume that we are given a finite class Γ_i of formulas which are conjunctions of clauses such that Γ_i is solution equivalent to $\phi(\bar{*})$. If some formula $\phi'(\bar{*}) \in \Gamma_i$ has a form

$$- - - \wedge (A_1 \wedge \cdots \wedge A_n \to B_1 \vee \cdots \vee B_m) \wedge - - -$$

with $m > 1$ we replace the formula in Γ_i by the set of formulas

$$- - - \wedge (A_1 \wedge \cdots \wedge A_n \to B_j) \wedge - - -$$

for $1 \leq j \leq m$. We obtain a new class Γ_{i+1} which is solution equivalent to $\phi(\bar{*})$. We repeat the process as long as Γ_{i+i_1} contains non Horn-clauses.

(iii): Elimination of predicate symbols. Assume that we are given a finite class Γ_k of formulas which are conjunctions of Horn clauses such that Γ_k is solution equivalent to $\phi(\bar{*})$. If some formula $\phi'(\bar{*}) \in \Gamma_k$ is not a SREU problem then one of the following cases must obtain:

– The formula $\phi'(\bar{*})$ has a form

$$- - - \wedge (\cdots \rightarrow \boldsymbol{p}(\bar{\boldsymbol{a}})) \wedge - - -$$

and the predicate symbol $\boldsymbol{p}$ does not occur in the antecedent of the clause. Then the formula is unsolvable as it can be always falsified in a suitable structure. We delete the formula from Γ_k.

– The formula $\phi'(\bar{*})$ has a form

$$- - - \wedge (\cdots \wedge \boldsymbol{q}(\bar{\boldsymbol{b}}) \wedge \cdots \rightarrow B) \wedge - - -$$

where B is either $\boldsymbol{p}(\bar{\boldsymbol{a}})$ and then $\boldsymbol{q}$ and $\boldsymbol{p}$ are distinct predicate symbols or B is an identity. Then we replace the formula in Γ_k by the formula

$$- - - \wedge (\cdots \wedge \cdots \rightarrow B) \wedge - - - .$$

– The formula $\phi'(\bar{*})$ has a form

$$- - - \wedge (\cdots \wedge \boldsymbol{p}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \wedge \cdots \rightarrow \boldsymbol{p}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)) \wedge - - - .$$

Then we replace the formula in Γ_k by two formulas

$$- - - \wedge (\cdots \wedge \cdots \rightarrow \boldsymbol{p}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))) \wedge - - - ,$$

$$- - - \wedge (\cdots \wedge \cdots \rightarrow \boldsymbol{a}_1 \doteq \boldsymbol{b}_1) \wedge \ldots \wedge (\cdots \wedge \cdots \rightarrow \boldsymbol{a}_n \doteq \boldsymbol{b}_n) \wedge - - - \ .$$

By the above changes we obtain a new class Γ_{k+1} which is solution equivalent to $\phi(\bar{*})$. We repeat the process as long as Γ_{k+k_1} contains formulas which are not SREU's.

3.7 Example. The above conversion is demonstrated with the formula

$$\exists x\, (p(a) \wedge p(b) \wedge (x \doteq a \vee x \doteq b) \rightarrow p(c)) \, .$$

Here a, b, and c are different constants. Note that the constant c is a solution of its 1-skeleton $p(a) \wedge p(b) \wedge (* \doteq a \vee * \doteq b) \rightarrow p(c)$. Converting the skeleton to a conjunction of clauses yields

$$(p(a) \wedge p(b) \wedge * \doteq a \rightarrow p(c)) \wedge (p(a) \wedge p(b) \wedge * \doteq b \rightarrow p(c)) \, .$$

Elimination of predicate symbols leads to four SREU problems:

$$\begin{array}{c} (* \doteq a \rightarrow a \doteq c) \wedge (* \doteq b \rightarrow a \doteq c) \, , \\ (* \doteq a \rightarrow a \doteq c) \wedge (* \doteq b \rightarrow b \doteq c) \, , \\ (* \doteq a \rightarrow b \doteq c) \wedge (* \doteq b \rightarrow a \doteq c) \, , \\ (* \doteq a \rightarrow b \doteq c) \wedge (* \doteq b \rightarrow b \doteq c) \, . \end{array} \qquad \text{(a)}$$

Problem (a) is the only solvable one as it is solved by the constant c.

4 Non-recursiveness of $\boldsymbol{Sk}_1(\psi)$

4.1 Language of arithmetic formulas in predicate calculus. We wish to simulate arithmetic by certain quantifier-free semiformulas of predicate calculus. The semiformulas will be in the language P consisting of constants $\mathbf{0}$, $\hat{\mathbf{0}}$, $\tilde{\mathbf{0}}$, k, $\tilde{k}$, of the unary function symbol S, and of the binary function symbol '$\boldsymbol{,}$'. We write the binary symbol in the infix form $\boldsymbol{a}\boldsymbol{,}\,\boldsymbol{b}$ where $\boldsymbol{a}\boldsymbol{,}\,\boldsymbol{b}\boldsymbol{,}\,\boldsymbol{c}$ associates to the right, i.e. it is read as $\boldsymbol{a}\boldsymbol{,}\,(\boldsymbol{b}\boldsymbol{,}\,\boldsymbol{c})$. The function symbol S will simulate the successor function, while the function symbol '$\boldsymbol{,}$' will play the role of a pairing function (*cons* of LISP).

We will define in Paragraphs 5.2, 5.7, and 5.16 quantifier-free semiformulas $Num(x)$, $Add(x,y,z,w)$, and $Mul(x,y,z,w,\tilde{w})$ of the language P with all of their free variables indicated. The semiformulas simulate arithmetic in predicate calculus as can be seen from the following lemma which will be proved in Sect. 5.

4.2 Lemma.

(a) *$Num(*)$ is solved exactly by $\mathbf{0}$-numerals,*
(b) *$Add(S^m(\mathbf{0}), S^p(\mathbf{0}), S^q(\mathbf{0}), *)$ is solvable iff $\mathcal{N} \models S^m(\mathbf{0}) + S^p(\mathbf{0}) \doteq S^q(\mathbf{0})$,*
(c) *$Mul(S^m(\mathbf{0}), S^p(\mathbf{0}), S^q(\mathbf{0}), *, *_1)$ is solvable iff $\mathcal{N} \models S^m(\mathbf{0}) \cdot S^p(\mathbf{0}) \doteq S^q(\mathbf{0})$.*

4.3 PC-arithmetic semiformulas. We simulate arithmetic by certain quantifier-free semiformulas of the language P where we use two disjoint sets of variables: $x_1, x_2, x_3, \ldots$ called *numeric* variables, and $w_1, w_2, w_3, \ldots$ called *table* variables. We will use $\boldsymbol{x}$ and $\boldsymbol{w}$ as metavariables ranging over numeric and table variables respectively. The semiformulas are built up from the semiformulas $Num(\boldsymbol{a})$, $Add(\boldsymbol{a},\boldsymbol{b},\boldsymbol{c},\boldsymbol{w})$, and $Mul(\boldsymbol{a},\boldsymbol{b},\boldsymbol{c},\boldsymbol{w},\tilde{\boldsymbol{w}})$ by conjunctions. Here the terms $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{c}$ are either $\mathbf{0}$-numerals or numeric variables.

We associate with every diophantine semiformula $\psi(\bar{\boldsymbol{x}})$ a quantifier-free semiformula $\phi(\bar{\boldsymbol{x}},\bar{\boldsymbol{w}})$ of P called a *PC-arithmetic semiformula*. The class of PC-arithmetic semiformulas is denoted by $\mathcal{A}$. The association is defined inductively as follows:

- $\boldsymbol{a}+\boldsymbol{b} \doteq \boldsymbol{c}$ is associated with any semiformula of the form
$$Num(\boldsymbol{a}) \wedge Num(\boldsymbol{b}) \wedge Num(\boldsymbol{c}) \wedge Add(\boldsymbol{a},\boldsymbol{b},\boldsymbol{c},\boldsymbol{w})\,,$$
- $\boldsymbol{a}\cdot\boldsymbol{b} \doteq \boldsymbol{c}$ is associated with any semiformula of the form
$$Num(\boldsymbol{a}) \wedge Num(\boldsymbol{b}) \wedge Num(\boldsymbol{c}) \wedge Mul(\boldsymbol{a},\boldsymbol{b},\boldsymbol{c},\boldsymbol{w}_1,\boldsymbol{w}_2)\,,$$
- a diophantine semiformula $\psi_1(\bar{\boldsymbol{x}}) \wedge \psi_2(\bar{\boldsymbol{x}})$ is associated with any semiformula of the form
$$\phi_1(\bar{\boldsymbol{x}},\bar{\boldsymbol{w}}_1) \wedge \phi_2(\bar{\boldsymbol{x}},\bar{\boldsymbol{w}}_2)\,,$$
where $\psi_1(\bar{\boldsymbol{x}})$ and $\psi_2(\bar{\boldsymbol{x}})$ are associated with $\phi_1(\bar{\boldsymbol{x}},\bar{\boldsymbol{w}}_1)$ and $\phi_2(\bar{\boldsymbol{x}},\bar{\boldsymbol{w}}_2)$ respectively and the table variables $\bar{\boldsymbol{w}}_1$ and $\bar{\boldsymbol{w}}_2$ are disjoint.

It is easy to see that if the diophantine semiformula $\psi(\bar{\boldsymbol{x}})$ is associated with $\phi(\bar{\boldsymbol{x}},\bar{\boldsymbol{w}}) \in \mathcal{A}$ then both semiformulas contain the same numeric variables and every numeric variable $\boldsymbol{x}$ of $\phi(\bar{\boldsymbol{x}},\bar{\boldsymbol{w}})$ occurs in a semiformula $Num(\boldsymbol{x})$.

4.4 Invariancy of association under substitution. We will use the following fact:

> if the diophantine semiformula $\psi(\boldsymbol{x}, \bar{\boldsymbol{x}})$ is associated with $\phi(\boldsymbol{x}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}) \in \mathcal{A}$
> then $\psi(S^m(\mathbf{0}), \bar{\boldsymbol{x}})$ is associated with $\phi(S^m(\mathbf{0}), \bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}) \in \mathcal{A}$

which is easily proved by induction on $\psi(\boldsymbol{x}, \bar{\boldsymbol{x}})$.

For the proof of the undecidability of 1-skeletons in Thm. 4.9 we need some auxiliary propositions.

4.5 Lemma. *If the diophantine formula ψ is associated with $\phi(\bar{\boldsymbol{w}}) \in \mathcal{A}$ then $\mathcal{N} \models \psi$ iff $\phi(\bar{*})$ is solvable.*

Proof. By induction on ψ. If ψ is $S^m(\mathbf{0}) + S^p(\mathbf{0}) \doteq S^q(\mathbf{0})$ then $\phi(\bar{*})$ has a form

$$Num(S^m(\mathbf{0})) \wedge Num(S^p(\mathbf{0})) \wedge Num(S^q(\mathbf{0})) \wedge Add(S^m(\mathbf{0}), S^p(\mathbf{0}), S^q(\mathbf{0}), *)\ .$$

The first three conjuncts are valid by Lemma 4.2(a) and the equivalence follows directly from Lemma 4.2(b). The case when ψ is $S^m(\mathbf{0}) \cdot S^p(\mathbf{0}) \doteq S^q(\mathbf{0})$ is similar and uses Lemma 4.2(c). If ψ is $\psi_1 \wedge \psi_2$ then $\phi(\bar{*})$ has a form $\phi_1(\bar{*}_1) \wedge \phi_2(\bar{*}_2)$ where $\bar{*}$ is partitioned into two disjoint sequences $\bar{*}_1$ and $\bar{*}_2$. Hence, $\mathcal{N} \models \psi$ iff $\mathcal{N} \models \psi_1$ and $\mathcal{N} \models \psi_2$ iff, by inductive hypotheses, $\phi_1(\bar{*}_1)$ and $\phi_2(\bar{*}_2)$ are solvable iff, because of disjointness of unknowns, $\phi_1(\bar{*}_1) \wedge \phi_2(\bar{*}_2)$ is solvable. $\square$

4.6 Lemma. *Let $\phi(\bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}) \in \mathcal{A}$ be a semiformula with all of its numeric variables indicated. If the formula $\phi(\bar{\boldsymbol{a}}, \bar{*})$ is solvable then the terms $\bar{\boldsymbol{a}}$ are $\mathbf{0}$-numerals.*

Proof. We recall (Par. 4.3) that every term $\boldsymbol{a}$ of $\bar{\boldsymbol{a}}$ is substituted for $\boldsymbol{x}$ in some conjunct $Num(\boldsymbol{x})$ of $\phi(\bar{\boldsymbol{x}}, \bar{\boldsymbol{w}})$. Thus if $\models \phi(\bar{\boldsymbol{a}}, \bar{\boldsymbol{b}})$ for some terms $\bar{\boldsymbol{b}}$ then for every term $\boldsymbol{a}$ of $\bar{\boldsymbol{a}}$ we have $\models Num(\boldsymbol{a})$ and by Lemma 4.2(a), the terms $\bar{\boldsymbol{a}}$ are $\mathbf{0}$-numerals. $\square$

4.7 Lemma. *Let the diophantine semiformula $\psi(\bar{\boldsymbol{x}})$ with all of its variables indicated be associated with some $\phi(\bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}) \in \mathcal{A}$. Then $\mathcal{N} \models \exists \bar{\boldsymbol{x}}\, \psi(\bar{\boldsymbol{x}})$ iff $\phi(\bar{*}, \bar{*}_1)$ is solvable.*

Proof. The diophantine semiformula $\psi(\bar{\boldsymbol{x}})$ has the same set of numeric variables as $\phi(\bar{\boldsymbol{x}}, \bar{\boldsymbol{w}})$. Thus $\mathcal{N} \models \exists \bar{\boldsymbol{x}}\, \psi(\bar{\boldsymbol{x}})$ iff $\mathcal{N} \models \psi(\bar{\boldsymbol{a}})$ for some $\mathbf{0}$-numerals $\bar{\boldsymbol{a}}$ iff, by Par. 4.4 and Lemma 4.5, $\phi(\bar{\boldsymbol{a}}, \bar{*}_1)$ is solvable for some $\mathbf{0}$-numerals $\bar{\boldsymbol{a}}$ iff, by Lemma 4.6 (in the direction $\Leftarrow$), $\phi(\bar{*}, \bar{*}_1)$ is solvable. $\square$

4.8 Theorem. *To every recursively enumerable predicate $R(m)$ there is a semiformula $\phi(\boldsymbol{x}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}) \in \mathcal{A}$ such that for all m*

$$R(m) \text{ iff } Sk_1(\exists\bar{\boldsymbol{x}}\exists\bar{\boldsymbol{w}}\, \phi(S^m(\mathbf{0}), \bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}))\,. \tag{a}$$

Proof. By the theorem of Matiyasevich there is a diophantine semiformula $\psi(\boldsymbol{x}, \bar{\boldsymbol{x}})$ with all free variables indicated such that for every number m

$$R(m) \text{ iff } \mathcal{N} \models \exists\bar{\boldsymbol{x}}\, \psi(S^m(\mathbf{0}), \bar{\boldsymbol{x}})\,. \tag{b}$$

Take a semiformula $\phi(\boldsymbol{x}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}) \in \mathcal{A}$ associated to $\psi(\boldsymbol{x}, \bar{\boldsymbol{x}})$. Then $R(m)$ holds iff, by (b), $\mathcal{N} \models \exists\bar{x}\, \psi(S^m(\mathbf{0}), \bar{x})$ iff, by Par. 4.4 and Lemma 4.7, $\phi(S^m(\mathbf{0}), \bar{*}, \bar{*}_1)$ is solvable iff $Sk_1(\exists\bar{\boldsymbol{x}}\exists\bar{\boldsymbol{w}}\, \phi(S^m(\mathbf{0}), \bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}))$. □

4.9 Theorem. *The predicate $Sk_1(\psi)$ is not recursive.*

Proof. Take any recursively enumerable but not recursive predicate $R(m)$ and obtain a semiformula $\phi(\boldsymbol{x}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}) \in \mathcal{A}$ from Thm. 4.8. We can clearly find a primitive recursive function f such that

$$f(m) = \exists\bar{\boldsymbol{x}}\exists\bar{\boldsymbol{w}}\, \phi(S^m(\mathbf{0}), \bar{\boldsymbol{x}}, \bar{\boldsymbol{w}})\,.$$

Then $R(m)$ iff $Sk_1(f(m))$ by 4.8(a). If the predicate $Sk_1(\psi)$ were recursive so would be $R(m)$. □

5 Simulation of Arithmetic

In this section we will define the semiformulas *Num*, *Add*, and *Mul* simulating arithmetic in predicate calculus and prove Lemma 4.2. This will finish the proof of the undecidability of Sk_1 (Thm. 4.9). The section is rather technical in that all proofs are carried out in detail. We do this on purpose in order to demonstrate that the problem of Herbrand skeletons is a purely logical problem albeit with extremely important consequences for ATP. Hence, we feel that the solution should be expressed in the well-developed apparatus of predicate calculus (see for instance [Sho67]) without any detours through the terminology and techniques of ATP and/or term rewriting. We start with a lemma which is used in ATP and term rewriting more or less automatically although its proof requires non-trivial properties of predicate calculus.

5.1 Lemma.

(a) *For a semiformula* $\phi(\boldsymbol{x})$ *with at most* $\boldsymbol{x}$ *free, term* $\boldsymbol{a}$*, and constant* $\boldsymbol{k}$ *occurring neither in* $\phi(\boldsymbol{x})$ *nor in* $\boldsymbol{a}$ *we have* $\models \boldsymbol{k} \doteq \boldsymbol{a} \to \phi(\boldsymbol{k})$ *iff* $\models \phi(\boldsymbol{a})$.

(b) $\models \boldsymbol{a} \doteq \boldsymbol{b}$ *iff* $\boldsymbol{a} = \boldsymbol{b}$.

Proof. (a): We have $\models \boldsymbol{k} \doteq \boldsymbol{a} \to \phi(\boldsymbol{k})$ iff, by the theorem on constants in [Sho67], $\models \forall \boldsymbol{x}(\boldsymbol{x} \doteq \boldsymbol{a} \to \phi(\boldsymbol{x}))$ iff, by the third corollary of the equality theorem in [Sho67], $\models \phi(\boldsymbol{a})$.

(b): Clearly, if $\boldsymbol{a} = \boldsymbol{b}$ then $\models \boldsymbol{a} \doteq \boldsymbol{b}$. For the reverse direction consider a structure $\mathcal{M}$ with the domain consisting of all terms (which are a subset of natural numbers) and with the interpretation I of function symbols such that $\boldsymbol{f}^I(\bar{d}) = f(\bar{d})$. Clearly, $\boldsymbol{c}^I = \boldsymbol{c}$ for all terms. If $\boldsymbol{a} \neq \boldsymbol{b}$ then $\boldsymbol{a}^I \neq \boldsymbol{b}^I$, i.e. $\mathcal{M} \models \boldsymbol{a} \neq \boldsymbol{b}$, and so $\not\models \boldsymbol{a} \doteq \boldsymbol{b}$. □

5.2 Numerals. Denote by $Num(x)$ the semiformula $\mathbf{0} \doteq S(\mathbf{0}) \to \mathbf{0} \doteq x$ and by $\widetilde{Num}(x)$ the semiformula $\tilde{\mathbf{0}} \doteq S(\tilde{\mathbf{0}}) \to \tilde{\mathbf{0}} \doteq x$.

5.3 Lemma.

(a) $Num(*)$ *is solved exactly by* $\mathbf{0}$*-numerals,*

(b) $\widetilde{Num}(*)$ *is solved exactly by* $\tilde{\mathbf{0}}$*-numerals.*

Proof. We prove only the part (a) as the proof of (b) is similar. $\models Num(S^n(\mathbf{0}))$ is proved by a straightforward induction on n. Conversely, if the term $\boldsymbol{a}$ is not a $\mathbf{0}$-numeral then it must be the case that $\boldsymbol{a} = S^n(\boldsymbol{f}(\bar{\boldsymbol{b}}))$ for a function symbol $\boldsymbol{f}$ different from $\mathbf{0}$ and S, some number n, and terms $\bar{\boldsymbol{b}}$. Consider a structure $\mathcal{M}$ with the domain $\{0, 1\}$ and the interpretation of function symbols I such that $\mathbf{0}^I = 0$, $S^I(d) = d$ for all d in the domain, and $\boldsymbol{g}^I(\bar{d}) = 1$, for all other symbols and all $\bar{d}$ in the domain. We clearly have $\mathbf{0}^I = 0 = (S(\mathbf{0}))^I$, i.e. $\mathcal{M} \models \mathbf{0} \doteq S(\mathbf{0})$ and also

$$(S^n(\boldsymbol{f}(\bar{\boldsymbol{b}})))^I = (S^I)^n((\boldsymbol{f}(\bar{\boldsymbol{b}}))^I) = (S^I)^n(1) = 1 \neq 0 = \mathbf{0}^I .$$

Hence, $\mathcal{M} \not\models \mathbf{0} \doteq S(\mathbf{0}) \to \mathbf{0} \doteq S^n(\boldsymbol{f}(\bar{\boldsymbol{b}}))$. Thus $\not\models Num(\boldsymbol{a})$. □

5.4 Proof of Lemma 4.2(a). This is Lemma 5.3(a). □

5.5 Similar numerals. Denote by $Sim(x, y)$ the semiformula $\mathbf{0} \doteq \tilde{\mathbf{0}} \to x \doteq y$.

5.6 Lemma. $\models Sim(S^m(\mathbf{0}), S^p(\tilde{\mathbf{0}}))$ *iff* $m = p$.

Proof. We have $\models Sim(S^m(\mathbf{0}), S^p(\tilde{\mathbf{0}}))$ iff $\models \mathbf{0} \doteq \tilde{\mathbf{0}} \to S^m(\mathbf{0}) \doteq S^p(\tilde{\mathbf{0}})$ iff, by Lemma 5.1(a), $\models S^m(\tilde{\mathbf{0}}) \doteq S^p(\tilde{\mathbf{0}})$ iff, by Lemma 5.1(b), $S^m(\tilde{\mathbf{0}}) = S^p(\tilde{\mathbf{0}})$ iff $m = p$. □

5.7 Addition. Denote by $\mathit{Plus}(x,y,z)$ the semiformula $\tilde{\mathbf{0}} \doteq x \to z \doteq y$ and by $\mathit{Add}(x,y,z,w)$ the semiformula

$$\widetilde{\mathit{Num}}(w) \wedge \mathit{Sim}(y,w) \wedge \mathit{Plus}(x,w,z)\,.$$

5.8 Lemma. $\models \mathit{Plus}(S^m(\mathbf{0}), S^p(\tilde{\mathbf{0}}), S^q(\mathbf{0}))$ *iff* $q = m + p$.

Proof. $\models \mathit{Plus}(S^m(\mathbf{0}), S^p(\tilde{\mathbf{0}}), S^q(\mathbf{0}))$ iff $\models \tilde{\mathbf{0}} \doteq S^m(\mathbf{0}) \to S^q(\mathbf{0}) \doteq S^p(\tilde{\mathbf{0}})$ iff, by Lemma 5.1(a), $\models S^q(\mathbf{0}) \doteq S^p(S^m(\mathbf{0}))$ iff, by Lemma 5.1(b), $S^q(\mathbf{0}) = S^p(S^m(\mathbf{0}))$ iff $q = m + p$. □

5.9 Proof of Lemma 4.2(b). $\mathit{Add}(S^m(\mathbf{0}), S^p(\mathbf{0}), S^q(\mathbf{0}), *)$ is solvable iff $\models \mathit{Add}(S^m(\mathbf{0}), S^p(\mathbf{0}), S^q(\mathbf{0}), \boldsymbol{d})$ for some $\boldsymbol{d}$ iff

$$\models \widetilde{\mathit{Num}}(\boldsymbol{d}) \wedge \mathit{Sim}(S^p(\mathbf{0}), \boldsymbol{d}) \wedge \mathit{Plus}(S^m(\mathbf{0}), \boldsymbol{d}, S^q(\mathbf{0}))$$

for some $\boldsymbol{d}$ iff, by Lemma 5.3(b),

$$\models \mathit{Sim}(S^p(\mathbf{0}), S^{p_1}(\tilde{\mathbf{0}})) \wedge \mathit{Plus}(S^m(\mathbf{0}), S^{p_1}(\tilde{\mathbf{0}}), S^q(\mathbf{0}))$$

for some p_1 iff, by Lemma 5.6, $\models \mathit{Plus}(S^m(\mathbf{0}), S^p(\tilde{\mathbf{0}}), S^q(\mathbf{0}))$ iff, by Lemma 5.8, $q = m + p$. □

5.10 Tables. Semiterms $\boldsymbol{a}(x,y,z)$ of the form

$$(S^{p_1}(x), S^{q_1}(y)), (S^{p_2}(x), S^{q_2}(y)), \ldots, (S^{p_r}(x), S^{q_r}(y)), z$$

are *semitables* of length $r \geq 0$. Note that the term z is a semitable of length 0. Closed instances of semitables are *tables*. Denote by $\mathit{Tab}(x)$ the semiformula

$$\mathbf{0} \doteq S(\mathbf{0}) \wedge k \doteq (\mathbf{0}, \mathbf{0}), k \to k \doteq x$$

and by $\widetilde{\mathit{Tab}}(x)$ the semiformula

$$\hat{\mathbf{0}} \doteq S(\hat{\mathbf{0}}) \wedge \tilde{\mathbf{0}} \doteq S(\tilde{\mathbf{0}}) \wedge \tilde{k} \doteq (\hat{\mathbf{0}}, \tilde{\mathbf{0}}), \tilde{k} \to \tilde{k} \doteq x\,.$$

5.11 Lemma.

(a) $\mathit{Tab}(*)$ *is solved exactly by* $\boldsymbol{a}(\mathbf{0}, \mathbf{0}, k)$ *where* $\boldsymbol{a}(x,y,z)$ *is a semitable,*
(b) $\widetilde{\mathit{Tab}}(*)$ *is solved exactly by* $\boldsymbol{a}(\hat{\mathbf{0}}, \tilde{\mathbf{0}}, \tilde{k})$ *where* $\boldsymbol{a}(x,y,z)$ *is a semitable.*

Proof. We prove only the part (b) as the proof of (a) is similar and even simpler. $\models \widetilde{\mathit{Tab}}(\boldsymbol{a}(\hat{\mathbf{0}}, \tilde{\mathbf{0}}, \tilde{k}))$ is proved by a straightforward induction on the length of the semitable $\boldsymbol{a}(x,y,z)$. Conversely, if $\models \mathit{Tab}(\boldsymbol{b})$ for a term $\boldsymbol{b}$ then also $\mathcal{M} \models \mathit{Tab}(\boldsymbol{b})$ for a structure $\mathcal{M}$ with the domain $\{0,2,3,4,5\}$ and the interpretation of function symbols I such that $\hat{\mathbf{0}}^I = 2$, $\tilde{\mathbf{0}}^I = 3$, $\tilde{k}^I = 4$, $S^I(2) = 2$, $S^I(3) = 3$, and $S^I(d) = 0$ for all other elements d of the domain. Furthermore, $2,^I 3 = 5$; $5,^I 4 = 4$; and

$d_1, {}^I d_2 = 0$ for all other elements d_1, d_2 of the domain. All other function symbols are interpreted as functions always yielding 0. Now, for the antecedent of $\widetilde{Tab}(\boldsymbol{b})$ we have

$$\mathcal{M} \models \hat{\mathbf{0}} \doteq S(\hat{\mathbf{0}}) \wedge \tilde{\mathbf{0}} \doteq S(\tilde{\mathbf{0}}) \wedge \tilde{k} \doteq (\hat{\mathbf{0}}, \tilde{\mathbf{0}}), \tilde{k}$$

and so $\mathcal{M} \models \tilde{k} \doteq \boldsymbol{b}$, i.e. $4 = \tilde{k}^I = \boldsymbol{b}^I$. This is clearly possible only if the term $\boldsymbol{b}$ is of a form $(\boldsymbol{b}_1, \boldsymbol{c}_1), (\boldsymbol{b}_2, \boldsymbol{c}_2), \ldots, (\boldsymbol{b}_n, \boldsymbol{c}_n), \tilde{k}$ for $n \geq 0$, $\boldsymbol{b}_j^I = 2$, and $\boldsymbol{c}_j^I = 3$ for all $1 \leq j \leq n$. By a reasoning similar to that in the proof of Lemma 5.3 we get that $\boldsymbol{b}_j$ and $\boldsymbol{c}_j$ are $\hat{\mathbf{0}}$ and $\tilde{\mathbf{0}}$-numerals respectively. Hence, $\boldsymbol{b} = \boldsymbol{a}(\hat{\mathbf{0}}, \tilde{\mathbf{0}}, \tilde{k})$ for a semitable $\boldsymbol{a}(x, y, z)$. □

5.12 Similar tables. Denote by $\widetilde{Sim}(x, y)$ the semiformula

$$\mathbf{0} \doteq \hat{\mathbf{0}} \wedge \mathbf{0} \doteq \tilde{\mathbf{0}} \wedge k \doteq \tilde{k} \rightarrow x \doteq y \,.$$

5.13 Lemma. *For semitables $\boldsymbol{a}(x, y, z)$ and $\boldsymbol{b}(x, y, z)$ we have*

$$\models \widetilde{Sim}(\boldsymbol{a}(\mathbf{0}, \mathbf{0}, k), \boldsymbol{b}(\hat{\mathbf{0}}, \tilde{\mathbf{0}}, \tilde{k})) \text{ iff } \boldsymbol{a}(x, y, z) = \boldsymbol{b}(x, y, z) \,.$$

Proof. We have $\models \widetilde{Sim}(\boldsymbol{a}(\mathbf{0}, \mathbf{0}, k), \boldsymbol{b}(\hat{\mathbf{0}}, \tilde{\mathbf{0}}, \tilde{k}))$ iff

$$\models \mathbf{0} \doteq \hat{\mathbf{0}} \wedge \mathbf{0} \doteq \tilde{\mathbf{0}} \wedge k \doteq \tilde{k} \rightarrow \boldsymbol{a}(\mathbf{0}, \mathbf{0}, k) \doteq \boldsymbol{b}(\hat{\mathbf{0}}, \tilde{\mathbf{0}}, \tilde{k})$$

iff, by Lemma 5.1, $\boldsymbol{a}(\mathbf{0}, \mathbf{0}, k) = \boldsymbol{b}(\mathbf{0}, \mathbf{0}, k)$ iff, by a straightforward induction on the length of the semitable $\boldsymbol{b}(x, y, z)$, we have $\boldsymbol{a}(x, y, z) = \boldsymbol{b}(x, y, z)$. □

5.14 (m, p)-tables. The tables we are interested in encode 'course of values' of multiplication and we call them (m, p)-*tables.* The term $\boldsymbol{a}(x, y, z)$ semitable is a $(m, 0)$-*semitable* if $\boldsymbol{a}(x, y, z) = z$. If $\boldsymbol{a}(x, y, z)$ is a (m, p)-semitable then the term $(S^p(x), S^{m \cdot p}(y)), \boldsymbol{a}(x, y, z)$ is a $(m, p+1)$-*semitable*. Note that (m, p)-semitables are of length p and they differ only in the variables x, y, and z. Closed instances of (m, p)-semitables are (m, p)-*tables*.

5.15 Lemma. *For a semitable $\boldsymbol{a}(x, y, z)$ we have*

$$\boldsymbol{a}(S(\mathbf{0}), S^m(\mathbf{0}), ((\mathbf{0}, \mathbf{0}), k)) = (S^p(\mathbf{0}), S^q(\mathbf{0})), \boldsymbol{a}(\mathbf{0}, \mathbf{0}, k) \tag{a}$$

iff $q = m \cdot p$ and $\boldsymbol{a}(x, y, z)$ is a (m, p)-semitable.

Proof. By induction on the length of the semitable $\boldsymbol{a}(x, y, z)$. If $\boldsymbol{a}(x, y, z) = z$ then (a) holds iff $(\mathbf{0}, \mathbf{0}), k = (S^p(\mathbf{0}), S^q(\mathbf{0})), k$ iff $p = 0$ and $q = 0$ iff $q = m \cdot p$ and z is a (m, p)-semitable.

If $\boldsymbol{a}(x, y, z) = (S^{p_1}(x), S^{q_1}(y)), \boldsymbol{a}_1(x, y, z)$ for some semitable $\boldsymbol{a}_1(x, y, z)$ then (a) holds iff, after some simplifications, we have

$$\boldsymbol{a}_1(S(\mathbf{0}), S^m(\mathbf{0}), ((\mathbf{0}, \mathbf{0}), k)) = (S^{p_1}(\mathbf{0}), S^{q_1}(\mathbf{0})), \boldsymbol{a}_1(\mathbf{0}, \mathbf{0}, k) \,,$$

and $S^{p_1+1}(\mathbf{0}), S^{m+q_1}(\mathbf{0}) = S^p(\mathbf{0}), S^q(\mathbf{0})$ iff, by IH,

$$q_1 = m \cdot p_1,\ \boldsymbol{a}_1(x,y,z) \text{ is a } (m,p_1)\text{-semitable, and } p_1+1 = p,\ m+q_1 = q. \tag{b}$$

From (b) we have that $\boldsymbol{a}(x,y,z)$ is a (m, p_1+1)-semitable, i.e. (m,p)-semitable, and

$$q = m + q_1 = m + m \cdot p_1 = m \cdot (p_1 + 1) = m \cdot p\,.$$

Vice versa, if $q = m \cdot p$ and $\boldsymbol{a}(x,y,z)$ is a (m,p)-semitable then from the last it must be the case that $p = p_1 + 1$, $q_1 = m \cdot p_1$, and that $\boldsymbol{a}_1$ is a (m,p_1)-semitable. Moreover,

$$q = m \cdot p = m \cdot (p_1 + 1) = m \cdot p_1 + m = q_1 + m$$

and (b) holds. □

5.16 Multiplication. Denote by $Tim(x,y,z,w,\tilde{w})$ the semiformula

$$\hat{\mathbf{0}} \doteq S(\mathbf{0}) \wedge \tilde{\mathbf{0}} \doteq x \wedge \tilde{k} \doteq (\mathbf{0},\mathbf{0}), k \to \tilde{w} \doteq (y,z), w$$

and by $Mul(x,y,z,w,\tilde{w})$ the semiformula

$$Tab(w) \wedge \widetilde{Tab}(\tilde{w}) \wedge Sim(w,\tilde{w}) \wedge Tim(x,y,z,w,\tilde{w})\,.$$

5.17 Lemma. *For a semitable $\boldsymbol{a}(x,y,z)$ we have*

$$\models Tim(S^m(\mathbf{0}), S^p(\mathbf{0}), S^q(\mathbf{0}), \boldsymbol{a}(\mathbf{0},\mathbf{0},k), \boldsymbol{a}(\hat{\mathbf{0}},\tilde{\mathbf{0}},\tilde{k}))$$

iff $q = m \cdot p$ and $\boldsymbol{a}(x,y,z)$ is a (m,p)-semitable.

Proof. We have $\models Tim(S^m(\mathbf{0}), S^p(\mathbf{0}), S^q(\mathbf{0}), \boldsymbol{a}(\mathbf{0},\mathbf{0},k), \boldsymbol{a}(\hat{\mathbf{0}},\tilde{\mathbf{0}},\tilde{k}))$ iff

$$\models \hat{\mathbf{0}} \doteq S(\mathbf{0}) \wedge \tilde{\mathbf{0}} \doteq S^m(\mathbf{0}) \wedge \tilde{k} \doteq (\mathbf{0},\mathbf{0}), k \to \boldsymbol{a}(\hat{\mathbf{0}},\tilde{\mathbf{0}},\tilde{k}) \doteq (S^p(\mathbf{0}), S^q(\mathbf{0})), \boldsymbol{a}(\mathbf{0},\mathbf{0},k)$$

iff, by Lemma 5.1, we have 5.15(a) iff, by Lemma 5.15, $q = m \cdot p$ and $\boldsymbol{a}(x,y,z)$ is a (m,p)-semitable. □

5.18 Proof of Lemma 4.2(c). We wish to prove that

$$Mul(S^m(\mathbf{0}), S^p(\mathbf{0}), S^q(\mathbf{0}), *, *_1)$$

is solvable iff $\mathcal{N} \models S^m(\mathbf{0}) \cdot S^p(\mathbf{0}) \doteq S^q(\mathbf{0})$. The proof is similar to that of Lemma 4.2(b) as given in Par. 5.9. □

5.19 Remark. Note that the formula $Add(S^m(\mathbf{0}), S^p(\mathbf{0}), S^{m+p}(\mathbf{0}), *)$ has a unique solution $S^p(\tilde{\mathbf{0}})$ which has the same size as $S^p(\mathbf{0})$ where under the *size* of a term we understand the number of its function symbols.

Note also that the formula $Mul(S^m(\mathbf{0}), S^p(\mathbf{0}), S^{m \cdot p}(\mathbf{0}), *, *_1)$ has a unique solution where both (m,p)-tables have the same size given by a polynomial in m and p.

6 Non-recursiveness of $Sk_n(\psi)$

The reader has surely noted that for every PC-arithmetic semiformula $\phi(\bar{\boldsymbol{x}}, \bar{\boldsymbol{w}})$ associated to a diophantine semiformula $\psi(\bar{\boldsymbol{x}})$ the formulas $\phi(\bar{*}, \bar{*}_1)$ are instances of SREU. Hence, the undecidability of $Sk_1(\psi)$ (see Thm. 4.9) can be proved also from the proof of the undecidability of SREU by Degtyarev and Voronkov [DV95a]. Actually, in a recent paper [DV95b], which cites an earlier version of the present paper, they gave an alternative proof of the undecidability of SREU by the reduction of Matiyasevich's result in a similar way as we did in the previous sections.

The undecidability of $Sk_n(\psi)$ for $n > 1$, which is our main result (Thm. 6.7), requires additional work. We now give an example illustrating why the generalization to $Sk_n(\psi)$ is not straightforward.

6.1 Example. Consider the semiformula $\phi(x, w) \in \mathcal{A}$ of the form

$$Num(x) \wedge Num(S(\mathbf{0})) \wedge Num(\mathbf{0}) \wedge Add(x, S(\mathbf{0}), \mathbf{0}, w)$$

which is associated to the diophantine semiformula $x + S(\mathbf{0}) \doteq \mathbf{0}$. We have $\mathcal{N} \not\models \exists x\, x + S(\mathbf{0}) \doteq \mathbf{0}$ and so the formula $\phi(*, *_1)$, which is a 1-skeleton of $\exists x \exists w\, \phi(x, w)$, is not solvable by Lemma 4.7. On the other hand, we can solve a 2-skeleton of the last formula because we have

$$\models \phi(\mathbf{0}, \tilde{\mathbf{0}}) \vee \phi(S(\mathbf{0}), S(\tilde{\mathbf{0}}))\,. \tag{a}$$

Indeed, (a) holds iff, after some simplifications which remove valid subformulas,

$$\models Sim(S(\mathbf{0}), \tilde{\mathbf{0}}) \vee Plus(S(\mathbf{0}), S(\tilde{\mathbf{0}}), \mathbf{0}) \tag{b}$$

iff

$$\models (\mathbf{0} \doteq \tilde{\mathbf{0}} \to S(\mathbf{0}) \doteq \tilde{\mathbf{0}}) \vee (\tilde{\mathbf{0}} \doteq S(\mathbf{0}) \to \mathbf{0} \doteq S(\tilde{\mathbf{0}}))$$

iff

$$\models \mathbf{0} \doteq \tilde{\mathbf{0}} \wedge \tilde{\mathbf{0}} \doteq S(\mathbf{0}) \to S(\mathbf{0}) \doteq \tilde{\mathbf{0}} \vee \mathbf{0} \doteq S(\tilde{\mathbf{0}}) \tag{c}$$

and the last is the case.

This is a typical situation where the solvability of a disjunction does not guarantee the solvability of disjuncts because their clauses may interfere through solutions (compare (b) to (c)) even if the disjuncts do not share the unknowns.

6.2 Variants of PC-arithmetic semiformulas. For each $i \geq 1$ denote by P_i the language consisting of function symbols S, '**,**' and of constants $\mathbf{0}_i$, $\hat{\mathbf{0}}_i$, $\tilde{\mathbf{0}}_i$, k_i, $\tilde{k}_i$. The constants of languages P_i are called *special* constants. For simplicity sake we use a different set of variables for each language P_i: $x^i_1, x^i_2, x^i_3, \ldots$. We use $\boldsymbol{x}_i$ and $\bar{\boldsymbol{x}}_i$ as metavariables ranging over variables and sequences of variables of P_i respectively. We do not distinguish between the numeric and table variables of PC-arithmetic semiformulas which will be written in the form $\phi(\bar{\boldsymbol{x}})$. The semiformula $\phi_i(\bar{\boldsymbol{x}}_i)$ of the language P_i is called a *variant* of a PC-arithmetic semiformula $\phi(\bar{\boldsymbol{x}})$ if it is obtained from the last semiformula by replacing each of its constants $\boldsymbol{k}$ by the special constant $\boldsymbol{k}_i$ and each of its variables x_j by x^i_j. Clearly, $\phi(*)$ is solvable iff $\phi_i(*)$ is. Moreover, the solutions differ only in the corresponding constants. We denote by $\mathcal{A}_i$ the class of semiformulas of P_i which are variants of semiformulas of $\mathcal{A}$.

For $n \geq 1$ we assign to every quantifier-free semiformula $\phi(\bar{\boldsymbol{x}}) \in \mathcal{A}$ the semiformula $\psi(\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_n)$ of the form

$$\phi_1(\bar{\boldsymbol{x}}_1) \wedge \cdots \wedge \phi_n(\bar{\boldsymbol{x}}_n)\,,$$

where every $\phi_i(\bar{\boldsymbol{x}}_i) \in \mathcal{A}_i$ for $1 \leq i \leq n$ is a variant of $\phi(\bar{\boldsymbol{x}})$. We denote by $\mathcal{A}^{(n)}$ the class of semiformulas obtained by this assignment from the class $\mathcal{A}$.

6.3 Invariancy of assignment under substitution. We will use the following fact which is similar to that in Par. 4.4:

> if for $n \geq 1$ the semiformula $\psi(\boldsymbol{x}_1, \bar{\boldsymbol{x}}_1, \ldots, \boldsymbol{x}_n, \bar{\boldsymbol{x}}_n) \in \mathcal{A}^{(n)}$ is assigned to the semiformula $\phi(\boldsymbol{x}, \bar{\boldsymbol{x}}) \in \mathcal{A}$ where $\boldsymbol{x}$ is a numeric variable then for every $\mathbf{0}$-numeral $S^m(\mathbf{0})$ the semiformula $\psi(S^m(\mathbf{0}_1), \bar{\boldsymbol{x}}_1, \ldots, S^m(\mathbf{0}_n), \bar{\boldsymbol{x}}_n) \in \mathcal{A}^{(n)}$ is assigned to the semiformula $\phi(S^m(\mathbf{0}), \bar{\boldsymbol{x}}) \in \mathcal{A}$

which is easily proved by induction on $\phi(\boldsymbol{x}, \bar{\boldsymbol{x}})$.

The Main theorem 6.7 on the undecidability of n-skeletons requires two lemmas the first of which will be proved in the next section.

6.4 Main lemma. *If for $n \geq 1$ the semiformulas $\phi_i(\bar{\boldsymbol{x}}_i) \in \mathcal{A}_i$ are variants of $\phi(\bar{\boldsymbol{x}}) \in \mathcal{A}$ for all $1 \leq i \leq n$ and*

$$\models \phi_1(\bar{\boldsymbol{a}}_1) \vee \cdots \vee \phi_n(\bar{\boldsymbol{a}}_n) \tag{a}$$

then $\models \phi_i(\bar{\boldsymbol{a}}_i)$ for some i.

Proof outline. If $\not\models \phi_i(\bar{\boldsymbol{a}}_i)$ for all $1 \leq i \leq n$ then we will construct a structure $\mathcal{M}$ falsifying the variants at the same time: $\mathcal{M} \not\models \phi_i(\bar{\boldsymbol{a}}_i)$ for all $1 \leq i \leq n$. We will then have $\mathcal{M} \not\models \phi_1(\bar{\boldsymbol{a}}_1) \vee \cdots \vee \phi_n(\bar{\boldsymbol{a}}_n)$ and (a) will not hold. We postpone the construction of $\mathcal{M}$ until Par. 7.10. $\square$

6.5 Lemma. *If for $n \geq 1$ the semiformula $\psi(\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_n) \in \mathcal{A}^{(n)}$ is assigned to $\phi(\bar{\boldsymbol{x}}) \in \mathcal{A}$ then*

$$Sk_1(\exists \bar{\boldsymbol{x}} \phi(\bar{\boldsymbol{x}})) \textit{ iff } Sk_n(\exists \bar{\boldsymbol{x}}_1 \ldots \exists \bar{\boldsymbol{x}}_n\, \psi(\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_n))\,.$$

Proof. ($\Rightarrow$): If $Sk_1(\exists \bar{\boldsymbol{x}}\, \phi(\bar{\boldsymbol{x}}))$ then $\phi(\boldsymbol{*})$ is solved by some terms $\bar{\boldsymbol{a}}$ of P and we can solve $\psi(\boldsymbol{*}_1, \ldots, \boldsymbol{*}_n)$ by the corresponding variants $\bar{\boldsymbol{a}}_i$. Thus also any n-skeleton of $\exists \bar{\boldsymbol{x}}_1 \ldots \exists \bar{\boldsymbol{x}}_n\, \psi(\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_n)$ is solvable.

($\Leftarrow$): If some n-skeleton of $\exists \bar{\boldsymbol{x}}_1 \ldots \exists \bar{\boldsymbol{x}}_n\, \psi(\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_n)$ is solvable then

$$\models \psi_1(\bar{\boldsymbol{a}}_{1,1}, \ldots, \bar{\boldsymbol{a}}_{1,n}) \vee \cdots \vee \psi_n(\bar{\boldsymbol{a}}_{n,1}, \ldots, \bar{\boldsymbol{a}}_{n,n})$$

for some terms $\bar{\boldsymbol{a}}_{j,k}$ $(1 \leq j, k \leq n)$. Distributing $\vee$'s over $\wedge$'s and weakening yields

$$\models \phi_1(\bar{\boldsymbol{a}}_{1,1}) \vee \cdots \vee \phi_n(\bar{\boldsymbol{a}}_{n,n})\,.$$

By the Main lemma 6.4 we have $\models \phi_j(\bar{\boldsymbol{a}}_{j,j})$ for some j. Hence, $\phi_j(\boldsymbol{*}_j)$ and also $\phi(\boldsymbol{*})$ are solvable. $\square$

6.6 Theorem. *Let $n \geq 1$. To every recursively enumerable predicate $R(m)$ there is a semiformula $\psi(\boldsymbol{x}_1, \bar{\boldsymbol{x}}_1, \ldots, \boldsymbol{x}_n, \bar{\boldsymbol{x}}_n) \in \mathcal{A}^{(n)}$ such that for all m*

$$R(m) \text{ iff } Sk_n(\exists \bar{\boldsymbol{x}}_1 \ldots \exists \bar{\boldsymbol{x}}_n\, \psi(S^m(\boldsymbol{0}_1), \bar{\boldsymbol{x}}_1, \ldots, S^m(\boldsymbol{0}_n), \bar{\boldsymbol{x}}_n))\,. \tag{a}$$

Proof. By Thm. 4.8 there is a PC-arithmetic semiformula $\phi(\boldsymbol{x}, \bar{\boldsymbol{x}}) \in \mathcal{A}$ such that for every number m

$$R(m) \text{ iff } Sk_1(\exists \bar{\boldsymbol{x}}\, \phi(S^m(\boldsymbol{0}), \bar{\boldsymbol{x}}))\,. \tag{b}$$

Take the semiformula $\psi(\boldsymbol{x}_1, \bar{\boldsymbol{x}}_1, \ldots, \boldsymbol{x}_n, \bar{\boldsymbol{x}}_n) \in \mathcal{A}^{(n)}$ assigned to $\phi(\boldsymbol{x}, \bar{\boldsymbol{x}}) \in \mathcal{A}$. Then $R(m)$ holds iff, by (b), $Sk_1(\exists \bar{\boldsymbol{x}}\, \phi(S^m(\boldsymbol{0}), \bar{\boldsymbol{x}}))$ iff, by Par. 6.3 and Lemma 6.5, $Sk_n(\exists \bar{\boldsymbol{x}}_1 \ldots \exists \bar{\boldsymbol{x}}_n\, \psi(S^m(\boldsymbol{0}_1), \bar{\boldsymbol{x}}_1, \ldots, S^m(\boldsymbol{0}_n), \bar{\boldsymbol{x}}_n))$. $\square$

6.7 Main theorem. *The predicate $Sk_n(\psi)$ is not recursive for any $n \geq 1$.*

Proof. We fix n, take any recursively enumerable but not recursive predicate $R(m)$, and obtain a semiformula $\psi(\boldsymbol{x}_1, \bar{\boldsymbol{x}}_1, \ldots, \boldsymbol{x}_n, \bar{\boldsymbol{x}}_n) \in \mathcal{A}^{(n)}$ from Thm. 6.6. We can clearly find a primitive recursive function f such that

$$f(m) = \exists \bar{\boldsymbol{x}}_1 \ldots \exists \bar{\boldsymbol{x}}_n\, \psi(S^m(\boldsymbol{0}_1), \bar{\boldsymbol{x}}_1, \ldots, S^m(\boldsymbol{0}_n), \bar{\boldsymbol{x}}_n)\,.$$

Then $R(m)$ iff $Sk_n(f(m))$ by 6.6(a). If the predicate $Sk_n(\psi)$ were recursive so would be $R(m)$. $\square$

6.8 Discussion. From Thm. 6.7 we can immediately see that for no n there can be a recursive function $f_n(\psi)$ which yields a bound on the size of solutions to n-skeletons of an existential formula ψ. Existence of such a function would make the non-recursive predicate $Sk_n(\psi)$ recursive by a simple test of all candidate solutions of the size less than the bound.

7 Proof of the Main Lemma

For the proof of the Main lemma 6.4 given in Par. 7.10 we must be able to falsify simultaneously n variants of a not valid formula simulating arithmetic in predicate calculus. For this we need a structure interpreting the special constants in such a way that the interpretations do not interfere with each other. Such structures form a family defined in the following paragraph.

7.1 Structures $\boldsymbol{\mathcal{M}_\alpha}$. We define a family of structures $\mathcal{M}_\alpha$ with natural numbers as domains. The structures are parameterized by functions $\alpha(\boldsymbol{k})$ interpreting the special constants into natural numbers.

We partition the domain of natural numbers into seven mutually disjoint subsets. This is done with the help of a pairing function

$$J(j,k) = (j+k)(j+k+1)+k+1$$

which is the standard pairing function as used in recursion theory (see for instance [Dav58, pg. 43]) but offset by one. Thus 0 is the only number not in the range of J and we have the *pairing* property that from $J(i,k) = J(i',k')$ we get $i = i'$ and $k = k'$. A computer scientist will realize that J can ve viewed as the function *cons* of LISP, 0 as *nil*, and the set of natural numbers as the set of S-expressions generated from the single atom *nil*.

The first set in the partition is the set $\{J(0,i) \mid i \in N\}$ and it plays the role of natural numbers, where the natural number i is embedded into $J(0,i)$. The next five sets are $\{J(j,i) \mid i \in N\}$ for $1 \le j \le 5$ and they will be used to interpret the special constants in a special way. The seventh set consists of the remaining natural numbers, i.e. of 0 and of the numbers of the form $J(j+6,i)$, and it will play no special role.

For a given function α we define the interpretation I of the structure $\mathcal{M}_\alpha$ as follows. All predicate symbols $\boldsymbol{p}$ are always false: $\boldsymbol{p}^I = \emptyset$. For a special constant $\boldsymbol{k}$ we have $\boldsymbol{k}^I = \alpha(\boldsymbol{k})$. Function symbols $\boldsymbol{f}$ other than the special constants, S, and '$\boldsymbol{,}$' are interpreted as functions always yielding 0: $\boldsymbol{f}^I(\bar{d}) = 0$. The interpretation of the unary function symbol S is as follows:

$$\begin{aligned} S^I\,J(0,i) &= J(0,i+1) \\ S^I\,J(1,i) &= J(1,i) \\ S^I\,J(2,i) &= J(2,i) \\ S^I\,J(3,i) &= J(3,i) \\ S^I(m) &= 0 \text{ otherwise.} \end{aligned}$$

We can see that the function S^I behaves as the successor function on the numbers of the form $J(0,i)$, while on the other numbers it has special properties.

The interpretation of the binary function symbol '$\boldsymbol{,}$' is as follows:

$$\begin{aligned} J(0,i)\boldsymbol{,}^I J(0,j) &= J(0,J(i,j)) \\ J(1,i)\boldsymbol{,}^I J(1,i) &= J(5,i) \end{aligned}$$

$$J(2,i) ,^I J(3,i) = J(5,i)$$
$$J(5,i) ,^I J(4,i) = J(4,i)$$
$$m ,^I n = 0 \text{ otherwise.}$$

We can see that the function '$,^I$' *reflects* J, i.e. behaves as a pairing function, on the numbers of the form $J(0,i)$.

The following simple property of structures will be often used in the proofs below.

7.2 Lemma. *Let $\mathcal{M}$ be a structure and let $\phi(\boldsymbol{x})$ be a semiformula with at most $\boldsymbol{x}$ free. If the terms $\boldsymbol{a}$ and $\boldsymbol{b}$ denote in $\mathcal{M}$ the same element, i.e. if $\mathcal{M} \models \boldsymbol{a} \doteq \boldsymbol{b}$, then $\mathcal{M} \models \phi(\boldsymbol{a}) \leftrightarrow \phi(\boldsymbol{b})$.*

Proof. See the lemma on Pg. 19 in [Sho67]. $\square$

7.3 Note. By the conventions on variants of PC-arithmetic semiformulas discussed in Par. 6.2 we denote by $Num_i(x) \in \mathcal{A}_i$ the corresponding variant of $Num(x) \in \mathcal{A}$. Similarly, for other semiformulas defined in Sect. 5. Lemmas proved in Sect. 5 hold also for the corresponding variants. To emphasize this we will, for instance, refer to Lemma 5.3$_i$ when we mean Lemma 5.3 modified for the variant $Num_i(x)$.

7.4 Lemma.

(a) $\models Num_i(\boldsymbol{a})$ *iff* $\mathcal{M}_\alpha \models Num_i(\boldsymbol{a})$ *where* $\alpha(\mathbf{0}_i) = J(1,i)$ *and* $\alpha(\boldsymbol{k}) \neq J(1,i)$ *for all other special constants,*

(b) $\models \widetilde{Num}_i(\boldsymbol{a})$ *iff* $\mathcal{M}_\alpha \models \widetilde{Num}_i(\boldsymbol{a})$ *where* $\alpha(\tilde{\mathbf{0}}_i) = J(3,i)$ *and* $\alpha(\boldsymbol{k}) \neq J(3,i)$ *for all other special constants.*

Proof. We prove only the part (a) as the proof of (b) is similar. The direction $\Rightarrow$ is obvious. For the direction $\Leftarrow$ we assume $\not\models Num_i(\boldsymbol{a})$ and then $\boldsymbol{a} = S^m\,\boldsymbol{f}(\bar{\boldsymbol{b}})$ for some m, $\bar{\boldsymbol{b}}$, and $\boldsymbol{f}$ which is neither S nor $\mathbf{0}_i$ by Lemma 5.3(a)$_i$. We can see similarly as in the proof of Lemma 5.3(a) that $\mathcal{M}_\alpha \models \mathbf{0}_i \doteq S(\mathbf{0}_i)$ and $\mathcal{M}_\alpha \not\models \mathbf{0}_i \doteq S^m\,\boldsymbol{f}(\bar{\boldsymbol{b}})$. Hence, $\mathcal{M}_\alpha \not\models Num_i(\boldsymbol{a})$. $\square$

7.5 Lemma. $\models Sim_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i))$ *iff* $\mathcal{M}_\alpha \models Sim_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i))$ *where* $\alpha(\mathbf{0}_i) = \alpha(\tilde{\mathbf{0}}_i) = J(0,0)$.

Proof. The direction $\Rightarrow$ is obvious. For the direction $\Leftarrow$ we assume $\mathcal{M}_\alpha \models Sim_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i))$ and, since $\mathbf{0}_i^I = J(0,0) = \tilde{\mathbf{0}}_i^I$, we have $\mathcal{M}_\alpha \models S^m(\tilde{\mathbf{0}}_i) \doteq S^p(\tilde{\mathbf{0}}_i)$ by Lemma 7.2. S^I behaves as the successor function on the numbers of the form $J(0,j)$ and thus it must be the case that $m = p$. We now get $\models Sim_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i))$ by Lemma 5.6$_i$. $\square$

7.6 Lemma. $\models \mathit{Plus}_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i), S^q(\mathbf{0}_i))$ *iff*

$$\mathcal{M}_\alpha \models \mathit{Plus}_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i), S^q(\mathbf{0}_i))$$

where $\alpha(\mathbf{0}_i) = J(0,0)$ *and* $\alpha(\tilde{\mathbf{0}}_i) = J(0,m)$.

Proof. The direction $\Rightarrow$ is obvious. For the direction $\Leftarrow$ we assume $\mathcal{M}_\alpha \models \mathit{Plus}_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i), S^q(\mathbf{0}_i))$, i.e.

$$\mathcal{M}_\alpha \models \tilde{\mathbf{0}}_i \doteq S^m(\mathbf{0}_i) \to S^q(\mathbf{0}_i) \doteq S^p(\tilde{\mathbf{0}}_i)\,.$$

We can easily see that $(S^m(\mathbf{0}_i))^I = J(0,m) = \tilde{\mathbf{0}}_i^I$ and hence $\mathcal{M}_\alpha \models S^q(\mathbf{0}_i) \doteq S^p(S^m(\mathbf{0}_i))$ by Lemma 7.2. By similar arguments as in the proof of Lemma 7.5 we can see that $q = m + p$. We now get $\models \mathit{Plus}_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i), S^q(\mathbf{0}_i))$ by Lemma 5.8$_i$. □

7.7 Lemma.

(a) $\models \mathit{Tab}_i(\boldsymbol{a})$ *iff* $\mathcal{M}_\alpha \models \mathit{Tab}_i(\boldsymbol{a})$ *where* $\alpha(\mathbf{0}_i) = J(1,i)$, $\alpha(k_i) = J(4,i)$, *and for all other special constants* $\boldsymbol{k}$ *we have* $\alpha(\boldsymbol{k}) \neq J(j+1,i)$ *for all* j,

(b) $\models \widetilde{\mathit{Tab}}_i(\boldsymbol{a})$ *iff* $\mathcal{M}_\alpha \models \widetilde{\mathit{Tab}}_i(\boldsymbol{a})$ *where* $\alpha(\hat{\mathbf{0}}_i) = J(2,i)$, $\alpha(\tilde{\mathbf{0}}_i) = J(3,i)$, $\alpha(\tilde{k}_i) = J(4,i)$, *and for all other special constants* $\boldsymbol{k}$ *we have* $\alpha(\boldsymbol{k}) \neq J(j+1,i)$ *for all* j.

Proof. We prove only the part (b) as the proof of (a) is similar and even simpler. The direction $\Rightarrow$ is obvious. For the direction $\Leftarrow$ we assume $\mathcal{M}_\alpha \models \widetilde{\mathit{Tab}}_i(\boldsymbol{a})$. Since

$$\mathcal{M}_\alpha \models \hat{\mathbf{0}}_i \doteq S(\hat{\mathbf{0}}_i) \wedge \tilde{\mathbf{0}}_i \doteq S(\tilde{\mathbf{0}}_i) \wedge \tilde{k}_i \doteq (\hat{\mathbf{0}}_i, \tilde{\mathbf{0}}_i), \tilde{k}_i$$

we must also have $\mathcal{M}_\alpha \models \tilde{k}_i \doteq \boldsymbol{a}$. Reasoning similar to that in the proof of Lemma 5.11(b) shows that $\boldsymbol{a} = \boldsymbol{b}(\hat{\mathbf{0}}_i, \tilde{\mathbf{0}}_i, \tilde{k}_i)$ for a semitable $\boldsymbol{b}(x,y,z)$. We now get $\models \widetilde{\mathit{Tab}}_i(\boldsymbol{a})$ by Lemma 5.11(b)$_i$. □

7.8 Lemma. *For semitables* $\boldsymbol{a}(x,y,z)$ *and* $\boldsymbol{b}(x,y,z)$ *we have*

$$\models \widetilde{\mathit{Sim}}_i(\boldsymbol{a}(\mathbf{0}_i, \mathbf{0}_i, k_i), \boldsymbol{b}(\hat{\mathbf{0}}_i, \tilde{\mathbf{0}}_i, \tilde{k}_i)) \tag{a}$$

iff $\mathcal{M}_\alpha \models \widetilde{\mathit{Sim}}_i(\boldsymbol{a}(\mathbf{0}_i, \mathbf{0}_i, k_i), \boldsymbol{b}(\hat{\mathbf{0}}_i, \tilde{\mathbf{0}}_i, \tilde{k}_i))$ *where*

$$\alpha(\mathbf{0}_i) = \alpha(\hat{\mathbf{0}}_i) = \alpha(\tilde{\mathbf{0}}_i) = \alpha(k_i) = \alpha(\tilde{k}_i) = J(0,0)\,.$$

Proof. The direction $\Rightarrow$ is obvious. For the direction $\Leftarrow$ we assume $\mathcal{M}_\alpha \models \widetilde{\mathit{Sim}}_i(\boldsymbol{a}(\mathbf{0}_i, \mathbf{0}_i, k_i), \boldsymbol{b}(\hat{\mathbf{0}}_i, \tilde{\mathbf{0}}_i, \tilde{k}_i))$. Since

$$\mathcal{M}_\alpha \models \mathbf{0}_i \doteq \hat{\mathbf{0}}_i \wedge \mathbf{0}_i \doteq \tilde{\mathbf{0}}_i \wedge k_i \doteq \tilde{k}_i$$

we obtain $\mathcal{M}_\alpha \models \boldsymbol{a}(\mathbf{0}_i, \mathbf{0}_i, k_i) \doteq \boldsymbol{b}(\mathbf{0}_i, \mathbf{0}_i, k_i)$ by Lemma 7.2. Note that all subterms of $\boldsymbol{a}(\mathbf{0}_i, \mathbf{0}_i, k_i)$ and $\boldsymbol{b}(\mathbf{0}_i, \mathbf{0}_i, k_i)$ denote numbers of the form $J(0,j)$ on which S and '**,**' behave as successor and pairing functions respectively. Thus by a straightforward induction on the semitable $\boldsymbol{b}(x,y,z)$ we get that $\boldsymbol{a}(x,y,z) = \boldsymbol{b}(x,y,z)$. We now get (a) by Lemma 5.13$_i$. □

7.9 Lemma. *For a semitable* $\boldsymbol{a}(x,y,z)$ *we have*

$$\models Tim_i(S^m(\mathbf{0}_i), S^p(\mathbf{0}_i), S^q(\mathbf{0}_i), \boldsymbol{a}(\mathbf{0}_i,\mathbf{0}_i,k_i), \boldsymbol{a}(\hat{\mathbf{0}}_i,\tilde{\mathbf{0}}_i,\tilde{k}_i)) \tag{a}$$

iff

$$\mathcal{M}_\alpha \models Tim_i(S^m(\mathbf{0}_i), S^p(\mathbf{0}_i), S^q(\mathbf{0}_i), \boldsymbol{a}(\mathbf{0}_i,\mathbf{0}_i,k_i), \boldsymbol{a}(\hat{\mathbf{0}}_i,\tilde{\mathbf{0}}_i,\tilde{k}_i)) \tag{b}$$

where $\alpha(\mathbf{0}_i) = \alpha(k_i) = J(0,0)$, $\alpha(\hat{\mathbf{0}}_i) = J(0,1)$, $\alpha(\tilde{\mathbf{0}}_i) = J(0,m)$, *and* $\alpha(\tilde{k}_i) = J(0, J(J(0,0),0))$.

Proof. The direction $\Rightarrow$ is obvious. For the direction $\Leftarrow$ we assume (b), i.e.

$$\begin{aligned}\mathcal{M}_\alpha \models \hat{\mathbf{0}}_i \doteq S(\mathbf{0}_i) \wedge \tilde{\mathbf{0}}_i \doteq S^m(\mathbf{0}_i) \wedge \tilde{k}_i \doteq (\mathbf{0}_i \boldsymbol{,} \mathbf{0}_i)\boldsymbol{,} k_i \to \\ \boldsymbol{a}(\hat{\mathbf{0}}_i,\tilde{\mathbf{0}}_i,\tilde{k}_i) \doteq (S^p(\mathbf{0}_i)\boldsymbol{,} S^q(\mathbf{0}_i))\boldsymbol{,} \boldsymbol{a}(\mathbf{0}_i,\mathbf{0}_i,k_i)\,.\end{aligned}$$

Since $\tilde{\mathbf{0}}_i^I = J(0,1) = S(\mathbf{0}_i)^I$, $\hat{\mathbf{0}}_i^I = J(0,m) = (S^m(\mathbf{0}_i))^I$, and

$$\tilde{k}_i^I = J(0, J(J(0,0),0)) = (\mathbf{0}_i \boldsymbol{,} \mathbf{0}_i)\boldsymbol{,} k)^I$$

we get

$$\mathcal{M}_\alpha \models \boldsymbol{a}(S(\mathbf{0}_i), S^m(\mathbf{0}_i), ((\mathbf{0}_i \boldsymbol{,} \mathbf{0}_i)\boldsymbol{,} k_i)) \doteq (S^p(\mathbf{0}_i)\boldsymbol{,} S^q(\mathbf{0}_i))\boldsymbol{,} \boldsymbol{a}(\mathbf{0}_i,\mathbf{0}_i,k_i) \tag{c}$$

by Lemma 7.2. Note that all subterms in (c) denote numbers of the form $J(0,j)$ on which S and '$\boldsymbol{,}$' behave as successor and pairing functions respectively. Thus

$$\boldsymbol{a}(S(\mathbf{0}_i), S^m(\mathbf{0}_i), ((\mathbf{0}_i \boldsymbol{,} \mathbf{0}_i)\boldsymbol{,} k_i)) = (S^p(\mathbf{0}_i)\boldsymbol{,} S^q(\mathbf{0}_i))\boldsymbol{,} \boldsymbol{a}(\mathbf{0}_i,\mathbf{0}_i,k_i)$$

and we have $q = m \cdot p$ and that $\boldsymbol{a}(x,y,z)$ is a (m,p)-semitable by Lemma 5.15_i from which we get (a) by Lemma 5.17_i. □

7.10 Proof of the Main lemma 6.4. For a given $n \geq 1$ and formulas $\phi_i(\bar{\boldsymbol{a}}_i)$ such that $\not\models \phi_i(\bar{\boldsymbol{a}}_i)$ for $1 \leq i \leq n$ we wish to find a structure $\mathcal{M}_\alpha$ falsifying all formulas: $\mathcal{M}_\alpha \not\models \phi_i(\bar{\boldsymbol{a}}_i)$. We construct such an interpretation of special constants α by stages.

We set $\alpha(\boldsymbol{k}) = 0$ for all constants $\boldsymbol{k}$ of languages P_{n+i+1}. The assignment of interpretations to the special constants of P_i $(1 \leq i \leq n)$ is such that they never receive interpretations $J(j+1,k)$ for any j and $k \neq i$. We now let i range from 1 through n. For each i exactly one of the following cases applies. The assignment α is constructed by stages such that in the stage i the special constants of the language P_i are assigned interpretation.

(i): Suppose that for a conjunct of $\phi_i(\bar{\boldsymbol{a}}_i)$ we have

$$\not\models Num_i(\boldsymbol{a}) \text{ or } \not\models Tab_i(\boldsymbol{a})\,.$$

Setting $\alpha(\mathbf{0}_i) = J(1,i)$, $\alpha(k_i) = J(4,i)$, and $\alpha(\tilde{\mathbf{0}}_i) = \alpha(\hat{\mathbf{0}}_i) = \alpha(\tilde{k}_i) = 0$ will cause at the end $\mathcal{M}_\alpha \not\models \phi_i(\bar{\boldsymbol{a}}_i)$ by Lemmas 7.4(a) or 7.7(a).

(ii): Suppose that **(i)** does not apply and that for a conjunct of $\phi_i(\bar{\boldsymbol{a}}_i)$ we have

$$\not\models \widetilde{Num}_i(\boldsymbol{a}) \text{ or } \not\models \widetilde{Tab}_i(\boldsymbol{a})\,.$$

Setting $\alpha(\hat{\mathbf{0}}_i) = J(2,i)$, $\alpha(\tilde{\mathbf{0}}_i) = J(3,i)$, $\alpha(\tilde{k}_i) = J(4,i)$, and $\alpha(\mathbf{0}_i) = \alpha(k_i) = 0$ will cause at the end $\mathcal{M}_\alpha \not\models \phi_i(\bar{\boldsymbol{a}}_i)$ by Lemmas 7.4(b) or 7.7(b).

(iii): Suppose that **(i)** and **(ii)** do not apply and that for a conjunct of $\phi_i(\bar{\boldsymbol{a}}_i)$ we have

$$\not\models \mathit{Sim}_i(S^m(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i)) \text{ or } \not\models \widetilde{\mathit{Sim}}_i(\boldsymbol{a}(\mathbf{0}_i, \mathbf{0}_i, k_i), \boldsymbol{b}(\hat{\mathbf{0}}_i, \tilde{\mathbf{0}}_i, \tilde{k}_i))\,.$$

Setting $\alpha(\boldsymbol{k}) = J(0,0)$ for all special constants $\boldsymbol{k}$ of P_i will cause at the end $\mathcal{M}_\alpha \not\models \phi_i(\bar{\boldsymbol{a}}_i)$ by Lemmas 7.5 or 7.8.

(iv): Suppose that **(i)**, **(ii)**, and **(iii)** do not apply. Then for a conjunct of $\phi_i(\bar{\boldsymbol{a}}_i)$ we must have

$$\not\models \mathit{Plus}_i(S^m(\mathbf{0}_i), S^p(\mathbf{0}_i), S^q(\mathbf{0}_i), S^p(\tilde{\mathbf{0}}_i)) \text{ or}$$
$$\not\models \mathit{Tim}_i(S^m(\mathbf{0}_i), S^p(\mathbf{0}_i), S^q(\mathbf{0}_i), \boldsymbol{a}(\mathbf{0}_i, \mathbf{0}_i, k_i), \boldsymbol{a}(\hat{\mathbf{0}}_i, \tilde{\mathbf{0}}_i, \tilde{k}_i))\,.$$

Setting $\alpha(\mathbf{0}_i) = \alpha(k_i) = J(0,0)$, $\alpha(\hat{\mathbf{0}}_i) = J(0,1)$, $\alpha(\tilde{\mathbf{0}}_i) = J(0,m)$, and $\alpha(\tilde{k}_i) = J(0, J(J(0,0),0))$ will cause at the end $\mathcal{M}_\alpha \not\models \phi_i(\bar{\boldsymbol{a}}_i)$ by Lemmas 7.6 or 7.9. $\square$

8 Conclusion

The negative solution of the problem of Herbrand skeletons has important consequences for ATP. Although ATP involves r.e. functions we wish to use a recursive proof function which either finds a proof or indicates reasons why a proof could not be found. It is not very pleasant to abort a proof search based on an r.e. function because the abort does not yield any indication as to what has caused the failure.

Our main result shows that it is not sufficient to specify the size of a skeleton, i.e. the number of existential axioms, one also needs a bound on the size of terms in a solution. Thus we are interested in finding as efficient as possible an algorithm which, given the size of the skeleton and a bound on the size of solutions, either finds a solution or reports a failure. It is hoped that from the failure we can then obtain an indication as to why the bounds were exceeded.

References

[Bib82] W. Bibel. *Automated Theorem Proving.* Vieweg Verlag, 1982.

[BJ80] G. S. Boolos and R. C. Jeffrey. *Computability and Logic.* Cambridge University Press, second edition, 1980.

[Bus95a] S. R. Buss. An introduction to proof theory. Typeset manuscript, to appear in *Handbook of Proof Theory*, 1995.

[Bus95b] S. R. Buss. On Herbrand's theorem. Typeset manuscript, to appear in Proceedings of LCC'95, 1995.

[Dav58] M. Davis. *Computability and Unsolvability.* McGraw-Hill, 1958.

[DV95a] A. Degtyarev and A. Voronkov. Simultaneous rigid E-unification is undecidable. UPMAIL Technical Report 105, Uppsala University, Computing Science Department, May 1995.

[DV95b] A. Degtyarev and A. Voronkov. Simultaneous rigid E-unification is undecidable. In *Proceedings of 8th Workshop, CSL'95*, 1995.

[Fit90] M. Fitting. *First-order Logic and Automated Theorem Proving.* Springer Verlag, 1990.

[GNPS88] J. H. Gallier, P. Narendran, D. Plaisted, and W. Snyder. Rigid E-unification is NP-complete. In *Proceedings, Third Annual Symposium on Logic in Computer Science*, pages 218–227, Edinburgh, Scotland, 1988. IEEE Computer Society.

[GNPS90] J. H. Gallier, P. Narendran, D. Plaisted, and W. Snyder. Rigid E-unification: NP-completeness and applications to equational matins. *Information and Computation*, 87(1/2):129–195, 1990.

[Hei67] J. van Heijenoort, editor. *From Frege to Gödel: A Source Book in Mathematical Logic, 1879-1931.* Harvard University Press, 1967.

[Her30] J. Herbrand. *Recherches sur la théorie de la démonstration.* PhD thesis, University of Paris, 1930. English translation of chapter 5 in [Hei67], pp. 525–581.

[HP93] P. Hájek and P. Pudlák. *Metamathematics of First-Order Arithmetic.* Springer Verlag, 1993.

[Kog95] Eric de Kogel. Rigid E-unification simplified. In *Proceedings of 4th International Workshop, TABLEAUX'95*, pages 17–30. Springer Verlag, 1995.

[KP88] J. Krajíček and P. Pudlák. The number of proof lines and the size of proofs in first order logic. *Arch. Math. Logic*, 27:69–84, 1988.

[Mat70] Yu. V. Matiyasevich. Recursively enumerable sets are diophantine. *Soviet Mathematical Doklady*, 11(2):354–358, 1970.

[Rob65] J. A. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12:23–41, 1965.

[Sho67] J. R. Shoenfield. *Mathematical Logic.* Addison-Wesley, 1967.