J. Symbolic Computation (1992) **14**, 437–457

A New Method for Undecidability Proofs of First Order Theories

RALF TREINEN

DFKI, Stuhlsatzenhausweg 3, W6600 Saarbrücken, Germany

(Received 21 March 1991)

We claim that the reduction of Post's Correspondence Problem to the decision problem of a theory provides a useful tool for proving undecidability of first order theories given by some interpretation. The goal of this paper is to define a framework for such reduction proofs. The method proposed is illustrated by proving the undecidability of the theory of a term algebra modulo the axioms of associativity and commutativity and of the theory of a partial lexicographic path ordering.

1. Introduction

The interest of this paper is twofold. First it proposes a general methodology for proving results of the kind:

> The first order theory of the predicate logic model $\mathcal{I} = \cdots$ is undecidable.

Second, besides examples that serve just for the illustration of the method proposed, we show some applications that are interesting in their own right.

We only consider theories of given models, in contrast to theories defined by some sets of axioms that are not necessarily complete (for instance the theory defined by the axioms of Boolean algebras is not complete). When applied to (the theory of) a given model $\mathcal{I}$, the method leads to an effective mechanism that yields for each instance P of the Post Correspondence Problem over the alphabet $\{a, b\}$ a formula, denoted $\underline{\texttt{solvable}}_P$, such that

$$P \text{ is solvable} \iff \mathcal{I} \models \underline{\texttt{solvable}}_P \tag{1.1}$$

Because of the effectiveness of the construction of this formula we immediately get the undecidability result for the theory from the well-known undecidability of Post's Correspondence Problem. Furthermore we are interested in showing not only undecidability of the whole theory of $\mathcal{I}$, but of a smallest possible fragment of this theory. In the construction of $\underline{\texttt{solvable}}_P$ we will therefore try to avoid alternations of quantifiers as far as possible.

The basic principle of the proof method proposed is the simulation of the two data types involved in Post's Correspondence Problem: strings and sequences (resp. sets). The

0747–7171/90/000000 + 00 $03.00/0

representation of the objects of these data types is performed by appropriate representation functions mapping the carrier sets into the universe of the model under consideration. The representation does not reflect directly in the theory of the model, especially there is no need for formulas characterizing the images of the representation functions. The operations on the data types are expressed by first order formulas that are to be designed in regard to the properties of the model.

The target formula $\underline{\texttt{solvable}}_P$ consists of a "frame" that is independent of the model under consideration but uses subformulas representing the operations on the data types. We present the frame formula and formulate the requirements that guarantee the "correctness" of the representation of the carrier sets and the pertaining operations.

To a large extent we constrain the meaning of the formulas only for those elements of the universe that represent objects of the data types. Moreover, beyond the correctness of data type representation we have to make sure that a certain relation on the universe is Noetherian. This is an inherent property of the model, since the well-foundedness of a relation is not expressible in first order logic.

Several other methods for proving undecidability of theories have been proposed in the literature. Tarski (1953) shows that a theory T is undecidable if some essentially undecidable and finitely axiomatizable theory T' (for instance the theory Q (Tarski *et al.* 1953a)) is relatively weakly interpretable in T. In order to show relative weak interpretability of T' in T one has to find first order formulas defining the universe and operations of T' in some consistent extension of T. Hence the correspondence between the theories is expressed completely within the logic.

The method of Rabin (1965) does not require a finite axiomatization of the underlying undecidable theory. Rabin (1965) summarizes his proof principle as follows: "If T' is an undecidable theory and T is a theory such that by using appropriate formulas of T to represent the universe of T' and the non-logical constants of T', every model of T' is obtained from some model of T, then T is also undecidable". In this way the translation of T' into T is again expressed in terms of first order logic, but in order to show the correctness of the translation it is necessary to prove the required correspondence of models.

The method proposed here takes a different point of view: It exploits the properties of the model instead of properties of the *theory of* the model. The logic is not involved in the definition of the representation functions: The intended applications concern universes constituting a formal language, such that the representation may often be performed on a purely symbolic level. The logic is only used in the realization of the operations of the data types. We will demonstrate some applications where this technique yields very simple reduction proofs.

A first set of applications illustrating the method proposed is concerned with equational problems, that is validity of formulas with equality as the only predicate symbol in the initial, respectively the free algebra of an equational specification (see Comon (1991) and Bürckert & Schmidt-Schauß (1989)). The first example (A) treats the decision problem for the theory of ground term algebras modulo the axioms of associativity and commutativity (AC for short) and has been given as an open problem in Comon (1988). In this paper the existential fragment has been shown decidable thus extending the results for AC unification (Stickel (1981), Livesey & Siekmann (1976), Fages (1987) and Kirchner (1985) for arbitrary additional free function symbols). This example shows that equational problems may be undecidable even in case that unification with free function symbols is decidable. The extension by the axiom of idempotency to ACI in Example (B) is straightforward.

A related result is the undecidability of the theory of ground terms modulo associativity alone (Example (F)). Quine (1946) showed already the undecidability of the theory of concatenation. He gives a translation of number theory to the theory of concatenation that yields a Σ_6-sentence for an instance of Hilbert's Tenth Problem, using the undecidability of Hilbert's Tenth Problem (Matijacevič (1970)) this proves the undecidability of the Σ_6-fragment of the theory of concatenation. On the other hand (Example (F)) shows the undecidability of the Σ_2-fragment of the theory of a ground term algebra modulo associativity. A unification algorithm for this theory (without free function symbols) has been given by Plotkin (1972). The decidability of the unification problem for term algebras modulo A has been conjectured in Plotkin (1972) and proven in Makanin (1977). Baader & Schulz (1991) show the decidability of the unification problem for associative functions together with free function symbols. In contrast to the AC case, associativity without commutativity is of unification type ω (see (Bürckert *et al.* 1989) for the classification of unification problems), this coincides with the observation that our technique yields undecidability of the Σ_3 fragment in the AC case but Σ_2 in the case of associativity.

The second field of application is the theory of ground terms equipped with some ordering relation. The undecidability of the "theory of subterm relation" has been shown in Venkataraman (1987) but without the extension to possibly infinite trees. Furthermore Venkataraman (1987) shows the decidability of the existential fragment. Here our interest lies in the comparison between Venkataraman's proof and ours. We extend our undecidability result to the case of infinite trees. The decidability of the the existential fragment of the theory of finite and infinite trees with the subtree relation was shown in Tulipani (1993).

The question of decidability of the theory of a total simplification ordering has been posed in Comon (1988). The decidability of the existential fragment of a total lexicographic path ordering (lpo for short) is shown in Comon (1990), the analogous result for a total recursive path ordering has been given in Jouannaud & Okada (1991). We prove in Example (C) the undecidability of the Σ_4 fragment of a partial lpo. Unfortunately there still remain two big gaps between these results (see Section 5).

The undecidability of the Σ_2 fragment of complete number theory (Example (G)) is of course by no means a new result; it is presented here merely for demonstrating some aspects of the method proposed. The undecidability of the Σ_1 fragment has been shown in Matijacevič (1970).

The separation of Post's Correspondence Problem into two datatypes induces the structure of the paper: After a survey of the mathematical framework in Section 2 the simulation of the data type "strings" is discussed in Section 3. In the applications this part will always be the trivial one. Section 4 describes the construction of the sentence $\underline{\texttt{solvable}}_P$ while presenting two alternative methods for the representation of construction sequences. In the first method sequences are viewed as sets. This method is easier to use than the second one representing sequences directly but is less powerful, since in some applications the second method can yield a smaller number of quantifier alternations in the formula $\underline{\texttt{solvable}}_P$.

2. Preliminaries

In this paper we consider unsorted first order logic where equality is not required. For the basic notions according syntax and semantics of first order logic the reader is referred to textbooks on mathematical logic, for instance Enderton (1972). We specify a predicate logic basis as a pair (P, F), where the set of function symbols F is given in the form $\langle f(n_f), g(n_g), \ldots\rangle$ and the set of predicate symbols $P = \langle\oplus(n_\oplus), \odot(n_\odot), \ldots\rangle$. The numbers in parentheses are not part of the syntax but indicate the arity of the symbols. If $=(2)$ is present in P it is always interpreted as equality. We will frequently use symbolic names for formulas, and in defining one formula we will often refer to other formulas via their symbolic "macro" names without giving an exact semantics of macro expansion for formulas. We only mention the following notations:

$w(x_1, \ldots, x_n)$ where w is a symbolic name for a formula stands for a formula the free variables of which are (possibly as proper subset) among $\{x_1, \ldots, x_n\}$. As usual $w(t_1, \ldots, t_n)$ denotes the result of simultaneously substituting in $w(x_1, \ldots, x_n)$ the x_i by the corresponding t_i. We write $\mathcal{I} \models w[r_1, \ldots, r_n]$ if w is satisfied in $\mathcal{I}$ by the assignment $\{x_i \leftarrow r_i\}$. For the sake of convenience we allow infix notion, for instance $(x)w(y)$ instead of $w(x, y)$. Furthermore, in the examples, we will sometimes use tuples of variables instead of a single variables. In this case of course we have to replace the corresponding quantifiers by quantifier strings of the same kind.

The set of formulas over a given basis is split up into fragments. Rogers (1987) defines the *number of quantifier alternations* of a formula in prenex normal form (Gallier (1986)) as "the number of pairs of adjacent but unlike quantifiers". If this number is n and the outermost quantifier is $\exists$ (resp. $\forall$) the formula belongs to the Σ_{n+1}- (resp. Π_{n+1}-) fragment. $\Sigma_0 = \Pi_0$ denotes the set of quantifier-free formulas. An arbitrary formula belongs to a certain fragment if it is logically equivalent to a prenex normal form formula contained in this fragment.

Given a set Σ of symbols, Σ^* denotes the set of finite and $\Sigma^+ = \Sigma^* \setminus \{\epsilon\}$ the set of finite nonempty strings over Σ. $\lhd$ is the prefix ordering on strings. An instance P of the *Post Correspondence Problem* over an alphabet Σ (Post (1946)) is given by a finite set of the form

$$\{(p_i, q_i) \mid 0 \leq i \leq m; p_i, q_i \in \Sigma^+\}$$

A *P-construction sequence* for $(u, v) \in \Sigma^* \times \Sigma^*$ is a sequence $\big((u_j, v_j)\big)_{j=1\ldots n}$ with $u_j, v_j \in \Sigma^*$ for all j, $u_1 = v_1 = \epsilon$, $u_n = u$ and $v_n = v$, and for each $1 \leq j \leq n-1$ there is a $0 \leq i \leq m$ with $u_{j+1} = u_j p_i$ and $v_{j+1} = v_j q_i$ where juxtaposition denotes the concatenation of strings. In this case (u, v) is called *P-constructible*. P is *solvable* if there is a $u \in \Sigma^+$ such that (u, u) is P-constructible. It is undecidable whether an instance of the Post Correspondence Problem is solvable (Post (1946)), provided Σ contains at least two elements.

Equational problems emerged from the study of unification problems that can now be considered as a special case of equational problems (see Siekmann (1989) for a survey on unification). For a set F of ranked function symbols let $T(F)$ denote the set of F-ground terms and $T(F, X)$ the set of F-terms that contain variables from the set X. $T(F)$ and $T(F, X)$ will also be considered as F-algebras where the symbols from F are given their Herbrand interpretation (Gallier (1986)). The basis and the model for equational problems are defined by an equational specification (F, E) in the sense of Ehrig & Mahr (1985). Here we consider the restriction to the one-sorted case, that is F

is a ranked set of function symbols and E is a set of implicitly universally quantified equations of F-terms. The only predicate symbol is the equality symbol, the set of function symbols is given by the specification. A *permutation equation* is an equation where all variables and function symbols have an equal number of occurrences on the left and right side respectively (Bürckert *et al.* 1989). The axioms of associativity and commutativity are an example of a set of permutation equations. Bürckert & Schmidt-Schauß (1989) designate the following models of a specification (F, E):

1 the *initial algebra* is the quotient of the ground term algebra $T(F)$ by the congruence generated by E.
2 the *E-free algebra* is the quotient of the term algebra $T(F, X)$ by the congruence generated by E where X is a not further specified infinite set of variables.

A discussion of term algebras can be found in Ehrig & Mahr (1985). In this context Bürckert & Schmidt-Schauß (1989) call the Π_3 fragment *special equational problems* and the Σ_2 fragment *special equational problems without independent parameters.*

The *lexicographic path ordering* on $T(F)$ has been described in Dershowitz (1987)[†] as a tool for proving termination of term rewriting systems. For a given partial order[‡] $<_F$ on the set F of function symbols the lexicographic path ordering $\preceq_{\text{lpo}}$ is recursively defined by

$$t = g(t_1, \ldots, t_n) \preceq_{\text{lpo}} f(s_1, \ldots, s_m) = s$$

iff $t = s$ or one of the following holds

1 $t \preceq_{\text{lpo}} s_i$ for some i
2 $g <_F f$ and $t_j \prec_{\text{lpo}} s$ for all j
3 $f = g$ and there is a $j \leq n$ with
   (a) $t_i = s_i$ for all $i < j$
   (b) $t_j \prec_{\text{lpo}} s_j$
   (c) $t_i \prec_{\text{lpo}} s$ for all $i > j$

where $x \prec_{\text{lpo}} y$ is an abbreviation for $x \preceq_{\text{lpo}} y \wedge x \neq y$. $\preceq_{\text{lpo}}$ is a simplification ordering (Dershowitz (1987)), especially it is a partial order containing the subterm ordering. $\preceq_{\text{lpo}}$ is total iff the underlying precedence $<_F$ is total.

In the context of ordering relations we will also consider algebras containing finite *and* infinite trees. Courcelle (1983) contains a treatment of infinite trees.

$f^n(t)$ means n applications of the unary function symbol f to the term t. $lth(s)$ denotes the length of the sequence s. □ designates the end of a proof, the end of an example will be marked by ◇.

3. Simulation of Strings

The first thing we need for the representation of the data type string is a coding function

$$\phi\colon \{a, b\}^* \to \mathcal{I}$$

[†] referring to an unpublished paper of Kamin and Lévy.

[‡] The definition in Dershowitz (1987) (precedence) is slightly more general in using quasi-orderings.

We will use the symbol ϕ also to denote the corresponding function $\phi\colon \{a,b\}^* \times \{a,b\}^* \to \mathcal{I}^2$. The operations that will be used in the simulation of Post's Correspondence Problem are the test for emptiness and for each single nonempty string a unary function that appends this fixed string to its argument. For the sake of generality this function will be represented as a formula instead of a term. More precisely, we need:

1 $\underline{\texttt{is-}\epsilon}(x)$
2 $(y)\underline{\texttt{is}}(x)\underline{v}$ for each $v \in \{a,b\}^+$

such that

[INJ] ϕ is injective
[EPS] For all $r \in \mathcal{I}$: $\mathcal{I} \models \underline{\texttt{is-}\epsilon}[r]$ iff $r = \phi(\epsilon)$
[CON] For all $r \in \mathcal{I}$, $v \in \{a,b\}^+$, $w \in \{a,b\}^*$: $\mathcal{I} \models [r]\underline{\texttt{is}}[\phi(w)]\underline{v}$ iff $r = \phi(wv)$

In applications we have to specify both ϕ and the formulas $\underline{\texttt{is-}\epsilon}$ and $\cdot\underline{\texttt{is}}\cdot\underline{v}$. This procedure contains a certain redundancy, an alternative is to give a different set of requirements on $\underline{\texttt{is-}\epsilon}$ and $\cdot\underline{\texttt{is}}\cdot\underline{v}$ such that the representation function can be derived from the definition of these formulas:

$$\begin{aligned}\phi(\epsilon) &:= \text{ the unique } r \text{ with } \mathcal{I} \models \underline{\texttt{is-}\epsilon}[r]\\ \phi(w) &:= \text{ the unique } r \text{ with } \mathcal{I} \models [r]\underline{\texttt{is}}[\phi(\epsilon)]\underline{w} \quad (w \neq \epsilon)\end{aligned}$$

We do not follow this line since it seems to be more natural to define the representation of strings explicitly. An advantage of this alternative way is that the requirements substituting [INJ], [EPS] and [CON] state only properties of the *theory of* the model instead of properties of the model itself. Anyway, with the next requirement we have no hope of staying within the scope of first order logic as has been explained in the introduction.

Definition. $\sqsubset$ is the relation on $\mathcal{I}$ defined by: $x \sqsubset y$ iff there is a $v \in \{a,b\}^+$ with $\mathcal{I} \models [y]\underline{\texttt{is}}[x]\underline{v}$. As usual $\sqsubset^*$ denotes the reflexive transitive closure of $\sqsubset$. Furthermore $\sqsubset$ generalizes to pairs by $(x_1,x_2) \sqsubset (y_1,y_2)$ iff $x_1 \sqsubset y_1$ and $x_2 \sqsubset y_2$.

If $r_1 = \phi(w_1)$ and $r_2 = \phi(w_2)$ then $r_1 \sqsubset r_2$ expresses the prefix relationship between w_1 and w_2. However the definition is not restricted to representatives of strings, we will need this definition and the pertaining requirement in its full generality later. The formula $\underline{\texttt{finite}}$ characterizes the set of elements of the universe where $\sqsubset$ is a Noetherian relation. This set has to contain *at least* (but may not be equal to) the image of ϕ.

[NOE] There is no infinite descending $\sqsubset$-chain $(r_i)_{i\geq 0}$ in $\mathcal{I}$ with $\mathcal{I} \models \underline{\texttt{finite}}[r_0]$.
[FIN] For all $w \in \{a,b\}^+$: $\mathcal{I} \models \underline{\texttt{finite}}[\phi(w)]$

Example 1. The basis B contains at least the function symbols $\epsilon(0), a(1), b(1)$ and the equality symbol $=(2)$. Let $\mathcal{I}$ be the algebra of B-ground terms modulo some set of equations that that is consistent with respect to $\{\epsilon, a, b\}$ (Dershowitz & Jouannaud (1990)). This means that different ground terms built only with the symbols ϵ, a and b have different interpretations in $\mathcal{I}$.

Deliberately confusing the characters a, b from the alphabet with the unary function symbols a, b we define

$$\begin{array}{rcl} \phi(\sigma_0 \cdots \sigma_n) & := & \sigma_n(\cdots(\sigma_0(\epsilon))\cdots) \\ \underline{\texttt{is-}\epsilon}(x) & := & x = \epsilon \\ (y)\underline{\texttt{is}}(x)\underline{\sigma_0 \cdots \sigma_n} & := & y = \sigma_n(\cdots(\sigma_0(x))\cdots) \\ \underline{\texttt{finite}}(x) & := & \textsc{True} \end{array}$$

The reader might easily check that these definitions fulfill all the requirements [INJ], [EPS], [CON], [NOE] and [FIN]. ◇

Example 2. The basis B contains at least the equality symbol $=(2)$ and the function symbols $\epsilon(0), f(1)$ and $+(n)$ with $n \geq 2$. Let $\mathcal{I}$ denote the algebra of B-ground terms modulo some set of permutation equations that have no occurrences of any function symbol but $+$. With the temporary definitions

$$\begin{array}{rcl} \overline{a}(t) & := & +(\epsilon, f(f(t), \epsilon, \ldots, \epsilon)) \\ \overline{b}(t) & := & +(f(\epsilon), f(f(t)), f(\epsilon), \ldots, f(\epsilon)) \end{array}$$

we define

$$\begin{array}{rcl} \phi(\sigma_0 \cdots \sigma_n) & := & \overline{\sigma_n}(\cdots(\overline{\sigma_0}(\epsilon))\cdots) \\ \underline{\texttt{is-}\epsilon}(x) & := & x = \epsilon \\ (y)\underline{\texttt{is}}(x)\underline{\sigma_0 \cdots \sigma_n} & := & y = \overline{\sigma_n}(\cdots(\overline{\sigma_0}(x))\cdots) \\ \underline{\texttt{finite}}(x) & := & \textsc{True} \end{array}$$

In the definitions of $\overline{a}(t)$ and $\overline{b}(t)$ the occurrences of ϵ, resp. $f(\epsilon)$ do the coding of the symbols a, resp. b. Since there are permutation equations for $+$, we use a doubled occurrence of f in order to distinguish the coding of t. The above definitions still constitute a correct representation of strings when we enlarge the model $\mathcal{I}$ to the free algebra $T(F, X)$ modulo E. ◇

The next example shows a nontrivial $\underline{\texttt{finite}}$ formula.

Example 3. B contains at least the function symbols $\epsilon(0), a(1), b(1)$ and the predicate symbols $=(2), \leq(2)$. Consider the algebra $\mathcal{I}$ of finite and infinite B-ground terms where $\leq$ is interpreted as the subterm relation.[†] We choose ϕ, $\underline{\texttt{is-}\epsilon}$ and $\cdot\underline{\texttt{is}}\cdot\underline{v}$ as in Example (1). The set of finite objects consists now of the terms built only with unary function symbols and containing the symbol ϵ.

$$\underline{\texttt{finite}}(x) \quad := \quad \epsilon \leq x \wedge \forall x'.x' \leq x \supset \left\{x' = \epsilon \vee \exists x''.x' = a(x'') \vee x' = b(x'')\right\}$$

If the set of non-unary function symbols $B' \subseteq B$ is finite we can transform the conclusion of the above implication into a Π_1-formula, thus saving one alternation of quantifiers:

$$\underline{\texttt{finite}}(x) := \epsilon \leq x \wedge \forall x'.x' \leq x \supset \bigwedge_{f \in B'} \forall \vec{z}.x' \neq f(\vec{z})$$

[†] Note that the case of finite terms only is covered by Example (1).

4. Solutions of P

We are now ready to define the subformula $\underline{\texttt{one-step}}_P$. The intended meaning of the formula $\underline{\texttt{one-step}}_P(y_1, y_2, y_3, y_4)$ is: "The pair of strings represented by (y_1, y_2) is obtained from the pair of strings represented by (y_3, y_4) by the application of one P-construction step." This is the only subformula that depends directly on the instance of the Post Correspondence Problem P:

$$\underline{\texttt{one-step}}_P(y_1, y_2, y_3, y_4) \quad := \quad \bigvee_{i=0,\ldots,m} ((y_1)\underline{\texttt{is}}(y_3)\underline{p_i} \wedge (y_2)\underline{\texttt{is}}(y_4)\underline{q_i})$$

where $P = \{(p_i, q_i) \mid i = 0, \ldots, m\}$.

4.1. SIMULATION OF SEQUENCES AS SETS

In order to construct the sentence $\underline{\texttt{solvable}}_P$ we have to formulate something like "there is a P-construction sequence such that $\cdots$". How can we express as a formula the fact that something represents a P-construction sequence? The key idea we are going to explore now is: Instead of talking directly about sequences we may view a P-construction sequence as a *set* of pairs of strings. Since by definition a P-construction sequence is strictly ordered by the prefix relation on (pairs of) strings we are able to recover the sequence from the set.

With this idea we can now define the subformula $\underline{\texttt{construction}}_P(x)$ meaning that x represents a P-construction sequence. $\underline{\texttt{construction}}_P$ uses the subformula $(y_1, y_2)\underline{\texttt{in}}(x)$ reflecting the element relationship, the definition of which depends again on the model under consideration. From now on let a fixed instance P of the Post Correspondence Problem be given.

$$\begin{align} \underline{\texttt{construction}}_P(x) := \forall y_1, y_2 . (y_1, y_2)\underline{\texttt{in}}(x) \supset & \notag \\ \{\underline{\texttt{is-}\epsilon}(y_1) \wedge \underline{\texttt{is-}\epsilon}(y_2)\} \vee & \tag{4.1} \\ \exists y_3, y_4 . (y_3, y_4)\underline{\texttt{in}}(x) \wedge \underline{\texttt{one-step}}_P(y_1, y_2, y_3, y_4) & \tag{4.2} \end{align}$$

Still leaving pending the definition of $\underline{\texttt{in}}$ we can now show

LEMMA 4.1. *For all $r_1, r_2, u, s \in \mathcal{I}$ with $(r_1, r_2) \sqsubset^* (u, u)$ and*

$$\begin{aligned} \mathcal{I} &\models \underline{\texttt{finite}}[u] \\ \mathcal{I} &\models \underline{\texttt{construction}}_P[s] \\ \mathcal{I} &\models [r_1, r_2]\underline{\texttt{in}}[s] \end{aligned}$$

If [INJ], [EPS], [CON] and [NOE] are fulfilled then $(r_1, r_2) \in \mathrm{I}\mathrm{M}(\phi) \times \mathrm{I}\mathrm{M}(\phi)$ and the associated pair of strings $\phi^{-1}(r_1, r_2)$ is P-constructible.

PROOF. We fix u and s with the above properties. Because of [NOE] there can not exist an infinite descending (w.r.t. $\sqsubset$) chain of pairs $(r_1, r_2) \sqsubset^* (u, u)$. We can therefore perform Noetherian induction on (r_1, r_2).

If $\mathcal{I} \models \underline{\texttt{is-}\epsilon}[r_1] \wedge \underline{\texttt{is-}\epsilon}[r_2]$ then [EPS] yields $(r_1, r_2) = \phi(\epsilon, \epsilon)$ and we are done.

Otherwise case (4.2) from the definition of $\underline{\texttt{construction}}_P$ applies, so there exist r_3, r_4 with $\mathcal{I} \models [r_3, r_4]\underline{\texttt{in}}[s]$ and $\mathcal{I} \models \underline{\texttt{one-step}}_P[r_1, r_2, r_3, r_4]$. From the definition of $\underline{\texttt{one-step}}_P$ follows $(r_3, r_4) \sqsubset (r_1, r_2) \sqsubset^* (u, u)$. The induction hypothesis yields that

$(r_3, r_4) \in \mathrm{I\scriptstyle M}(\phi) \times \mathrm{I\scriptstyle M}(\phi)$ and $\phi^{-1}(r_3, r_4)$ is P-constructible, and because of [CON] and the definition of $\underline{\texttt{one-step}}_P$ the same holds for (r_1, r_2). □

We are now ready to define $\underline{\texttt{solvable}}_P$.

$$\underline{\texttt{solvable}}_P \quad := \quad \exists x, y. \underline{\texttt{construction}}_P(x) \wedge \underline{\texttt{finite}}(y) \wedge (y, y)\underline{\texttt{in}}(x) \wedge \neg \underline{\texttt{is-}\epsilon}(y)$$

From the above lemma we get immediately

Corollary 4.1. *If [INJ], [EPS], [CON] and [NOE] are fulfilled then*

$$\mathcal{I} \models \underline{\texttt{solvable}}_P \quad \Longrightarrow \quad P \textit{ is solvable}$$

The reader should note that up to now we did not need any constraints on the subformula $\underline{\texttt{in}}$. We made use of the special properties of the model only in order to fulfill the requirements in connection with the simulation of strings. Once the representation of strings with the subformulas $\underline{\texttt{is-}\epsilon}$, $\cdot\underline{\texttt{is}} \cdot \underline{v}$, $\underline{\texttt{finite}}$ is found, we get the first direction of our ”goal”–theorem (1.1) for free — that is without worrying about the representation of sequences.

In order to prove the opposite direction of (1.1) we now have to choose a representation function for P-construction sequences and a corresponding formula $(y_1, y_2)\underline{\texttt{in}}(x)$. S denotes the domain of the representation function $\psi \colon S \to \mathcal{I}$:

$$S := \{(u_i, v_i)_{i=1\ldots n} \mid u_i, v_i \in \{a, b\}^*, n \geq 2, u_i \lhd u_{i+1}, v_i \lhd v_{i+1}, (u_1, v_1) = (\epsilon, \epsilon)\}$$

Our last requirement relates the representation function ψ with the subformula $\underline{\texttt{in}}$ that is supposed to express the element relationship:

[IN] For all $r_1, r_2 \in \mathcal{I}$, $s \in S$: $\mathcal{I} \models [r_1, r_2]\underline{\texttt{in}}[\psi(s)]$ iff there exits $j \in \{1, \ldots, lth(s)\}$ with $(r_1, r_2) = \phi(s(j))$

Lemma 4.2. *If [EPS], [CON], [FIN], [IN] are fulfilled then*

$$P \textit{ is solvable} \quad \Longrightarrow \quad \mathcal{I} \models \underline{\texttt{solvable}}_P$$

The next theorem summarizes the method as it stands now:

Theorem 4.1. *Let B be a predicate logic basis and $\mathcal{I}$ a model for B. If we can find representation functions ϕ, ψ and formulas $\underline{\texttt{is-}\epsilon}$, $\cdot\underline{\texttt{is}}\cdot\underline{v}$, $\underline{\texttt{finite}}$, $\underline{\texttt{in}}$ such that [INJ], [EPS], [CON], [FIN], [NOE] and [IN] are fulfilled then the first order theory of $\mathcal{I}$ is undecidable.*

Now we can complete the examples started in Section 3:

Example A. Consider equational problems for the equational specification $(F_A, AC(+))$ where $F_A := \langle \epsilon(0), a(1), b(1), f(2), +(2) \rangle$ and $AC(+)$ denotes the axioms of associativity and commutativity for $+$:

$$\begin{aligned} x + y &= y + x \\ (x + y) + z &= x + (y + z) \end{aligned}$$

We take the representation of strings from Example (1). It is easy to see that with the following definitions [IN] is fulfilled in the initial and in the free algebra:

$$\begin{array}{rcl} \psi((u_i,v_i)_{i=1,\ldots,n}) & := & f(\phi(u_1),\phi(v_1)) + \cdots + f(\phi(u_n),\phi(v_n)) \\ (y_1,y_2)\underline{\mathtt{in}}(x) & := & \exists x'.x = f(y_1,y_2) + x' \end{array}$$

We can improve this result by restricting the base to $F_{A'} := \langle \epsilon(0), f(1), +(2)\rangle$ and $P_{A'} := \langle =(2)\rangle$. With the representation of strings as in Example (2) and the following definitions:

$$\begin{array}{rcl} \psi((u_i,v_i)_{i=1,\ldots,n}) & := & f(f(\phi(u_1)) + f(f(\phi(v_1)))) + \cdots + f(f(\phi(u_n)) + f(f(\phi(v_n)))) \\ (y_1,y_2)\underline{\mathtt{in}}(x) & := & \exists x'.x = f(f(y_1) + f(f(y_2))) + x' \end{array}$$

we obtain undecidability of the first order theory of $T(F_{A'})/_{AC(+)}$ and of the first order theory of $T(F_{A'},X)/_{AC(+)}$. In this construction the AC operator $+$ is used as a set constructor *and* as a string constructor. This is made possible by the insertion of the free function symbol f that serves as a barrier between the different occurrences of $+$ in the representation of strings. Besides this f is used to build distinguishable items in the representation of strings and in order to identify the two components of a pair of strings in the definition of ψ.

With an analogous construction we can show that undecidability holds in the case of $\langle \epsilon(0), f(n), +(2)\rangle$ with $n \geq 1$ and $+$ associative and commutative, and also in the case of the signature $\langle \epsilon(0), *(2), +(2)\rangle$ where both $+$ and $*$ are associative and commutative.

THEOREM 4.2. *The (Σ_3 fragment of the) first order theory of a ground term algebra (resp. term algebra) modulo associativity and commutativity is undecidable for signatures that contain at least one constant, one non-constant function symbol and one binary AC function symbol.* ◇

EXAMPLE B. Theorem 2 still holds when the set of axioms is enlarged by the axiom of idempotency:

$$x + x = x$$

THEOREM 4.3. *The (Σ_3 fragment of the) first order theory of a ground term algebra (resp. term algebra) modulo associativity, commutativity and idempotency is undecidable.*

It is easily seen that the above construction applies here also. The correctness of the representation of strings has to be checked separately since the axiom of idempotency is not a permutation equation. ◇

EXAMPLE C. Let $F_C := \langle \epsilon(0), a(1), b(1), e(1), l(1), h(3)\rangle$ and $P_C := \langle =(2), \leq(2)\rangle$. $\mathcal{I}_C$ is the ground term algebra $T(F_C)$ where $\leq$ is interpreted as the lexicographic path ordering $\preceq_{\mathrm{lpo}}$ generated by the following precedence on F_C:

$$\epsilon <_F a <_F b <_F h <_F \left\{ \begin{array}{l} l \\ e \end{array} \right.$$

e and l are incomparable in the order $<_F$.

ϕ, $\underline{\mathtt{is-}\epsilon}$, $\underline{\mathtt{finite}}$ and $\cdot\underline{\mathtt{is}}\cdot\underline{v}$ can be copied from Example (1). A P construction sequence will be represented by two lists of labeled strings, one for the first component and one

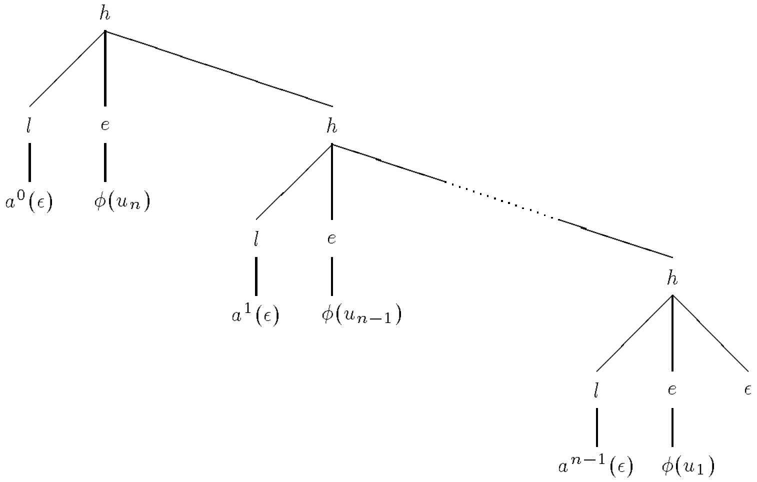

Figure 1. The term $\delta((u_i)_{i=1\ldots n})$ representing the sequence $(u_i)_{i=1\ldots n}$

for the second. The labels associate the corresponding components of a pair, moreover they will be essential for the formulation of the membership relation.

We associate to each nonempty sequence $s = (u_i)_{i=1,\ldots,n}$ the term $\delta(s)$ as shown in Figure 1 and choose

$$\psi((u_i, v_i)_{i=1,\ldots,n}) := (\delta((u_i)_{i=1,\ldots,n}), \delta((v_i)_{i=1,\ldots,n}))$$

In order to formulate the subformula $\underline{\texttt{in}}$ we use the following temporary definition:

$$\begin{aligned}(y)\underline{\texttt{in}}(x)\underline{\texttt{at}}(z) \quad := \quad & h(l(z), e(y), \epsilon) \leq x \wedge & (4.3)\\ & \forall y'.h(l(z), e(y'), \epsilon) \leq x \supset y' \leq y & (4.4)\end{aligned}$$

Finally we define

$$(y_1, y_2)\underline{\texttt{in}}(x_1, x_2) \quad := \quad \exists z.(y_1)\underline{\texttt{in}}(x_1)\underline{\texttt{at}}(z) \wedge (y_2)\underline{\texttt{in}}(x_2)\underline{\texttt{at}}(z)$$

THEOREM 4.4. *The (Σ_4 fragment of the) first order theory of a partial lexicographic path ordering is undecidable.*

The proof of [IN] is given in Appendix A.1. The separation of the P-construction sequence into two lists is not essential for the proof. In fact an analogous proof where the P-construction sequence is represented by one list of pairs of strings is also possible (by changing the arity of h to 4). The price of this variant is the need for another maximal function symbol incomparable to e and l, thereby leading to a “less total” ordering. In this alternative proof the labels $l(a^i(\epsilon))$ can not be omitted, they are necessary for the maximality condition in the definition of $\underline{\texttt{in}}$.

We remark that the same construction can be used to show that the Σ_4 fragment of the first order theory of a partial recursive path ordering (Dershowitz (1982)) is undecidable. ◇

Counting the quantifiers involved in the above construction we find that the formula

$\underline{\texttt{solvable}}_P$ is at least in the Σ_3 fragment. This is an inherent drawback of this method since $\underline{\texttt{solvable}}_P$ follows the pattern

$$\exists s \cdots \forall (s_1, s_2) \in s \cdots \exists (s_3, s_4) \in s \cdots$$

In general the formula $\underline{\texttt{in}}$ is the most "expensive" one (in terms of alternations of quantifiers). We will always try to find a formula $\underline{\texttt{in}}$ in Σ_1, if we do not succeed we get undecidability only for a fragment larger than Σ_3.

4.2. DIRECT SIMULATION OF SEQUENCES

In some applications it is possible to overcome this limitation by using a direct simulation technique for sequences. In this case we have to perform three different operations on the data type sequence, and we have to work a little bit harder to regulate the correlation of the pertaining formulas. We will come back to a comparison of these two methods at the end of this section.

The formulas that are to be designed for the model under consideration are

1 $\underline{\texttt{nonempty}}(x)$
2 $(y_1, y_2, x')\underline{\texttt{init-of}}(x)$
3 $(y_1, y_2)\underline{\texttt{head-of}}(x)$

The intended meaning of the first formula should be clear. It is useful to consider sequences as being constructed from right to left. Using this point of view, we call the element with the highest index in a sequence the *head* of the sequence. $(y_1, y_2)\underline{\texttt{head-of}}(x)$ is intended to express that (y_1, y_2) is the head of the sequence x. $(y_1, y_2, x')\underline{\texttt{init-of}}(x)$ is supposed to express that the sequence consisting of the sequence x' plus the head (y_1, y_2) is an initial segment of the sequence x.

The analogous definition of $\underline{\texttt{construction}}_P$ and $\underline{\texttt{solvable}}_P$ are given below. Note that, in contrast to Section 4.1, we now use an universal quantifier instead of an existential quantifier inside the definition of $\underline{\texttt{construction}}_P$.

$$\begin{aligned} &\underline{\texttt{construction}}_P(x) := \forall y_1, y_2, x'.(y_1, y_2, x')\underline{\texttt{init-of}}(x) \supset \\ &\quad \{\underline{\texttt{is-}\epsilon}(y_1) \wedge \underline{\texttt{is-}\epsilon}(y_2)\} \vee \\ &\quad \{\underline{\texttt{nonempty}}(x') \wedge \forall y_3, y_4.(y_3, y_4)\underline{\texttt{head-of}}(x') \supset \underline{\texttt{one-step}}_P(y_1, y_2, y_3, y_4)\} \end{aligned}$$

$$\underline{\texttt{solvable}}_P := \exists x, y.\underline{\texttt{construction}}_P(x) \wedge (y, y)\underline{\texttt{head-of}}(x) \wedge \underline{\texttt{finite}}(y) \wedge \neg\underline{\texttt{is-}\epsilon}(y)$$

In contrast to Section 4.1 where we obtained the first direction of (1.1) just from the properties of the representation of strings, we now have to state additional requirements on the newly introduced formulas:

[NH] $\mathcal{I} \models \forall x.\underline{\texttt{nonempty}}(x) \supset \exists y_1, y_2.(y_1, y_2)\underline{\texttt{head-of}}(x)$
[HS] $\mathcal{I} \models \forall x, y_1, y_2.(y_1, y_2)\underline{\texttt{head-of}}(x) \supset \exists x'.(y_1, y_2, x')\underline{\texttt{init-of}}(x)$
[HSH] $\mathcal{I} \models \forall x, x', y_1, y_2, y_3, y_4.(y_1, y_2, x')\underline{\texttt{init-of}}(x) \wedge (y_3, y_4)\underline{\texttt{head-of}}(x') \supset$
$\exists x''.(y_3, y_4, x'')\underline{\texttt{init-of}}(x)$

At this point the reader might remark that we could have used [NH] as a definition of $\underline{\texttt{nonempty}}$ by turning the implication sign into an equivalence. In this case only the requirement [HS] and [HSH] remain relating $\underline{\texttt{head-of}}$ to $\underline{\texttt{init-of}}$. We do not choose this approach in order to avoid the introduction of extra quantifiers. Example (D) shows how a model specific argument leads to the elimination of an unwanted existential quantifier in the definition of $\underline{\texttt{nonempty}}$.

With the help of these properties we can now prove a lemma analogous to Lemma 4.1:

Lemma 4.3. *For all* $r_1, r_2, u, s, s' \in \mathcal{I}$ *with* $(r_1, r_2) \sqsubset^* (u, u)$ *and*

$$\begin{aligned} &\mathcal{I} \models \underline{\texttt{finite}}[u] \\ &\mathcal{I} \models \underline{\texttt{construction}_P}[s] \\ &\mathcal{I} \models [r_1, r_2, s']\underline{\texttt{init-of}}[s] \end{aligned}$$

If [INJ], [EPS], [CON], [NOE], [NH] and [HSH] are fulfilled then $(r_1, r_2) \in \mathrm{Im}(\phi) \times \mathrm{Im}(\phi)$ *and* $\phi^{-1}(r_1, r_2)$ *is* P*-constructible.*

Proof. As in the proof of Lemma 4.1 we proceed by Noetherian induction on (r_1, r_2).

If $\mathcal{I} \models \underline{\texttt{is-}\epsilon}[r_1] \wedge \underline{\texttt{is-}\epsilon}[r_2]$ we know from [EPS] that $(r_1, r_2) = \phi(\epsilon, \epsilon)$.

Otherwise $\mathcal{I} \models \underline{\texttt{nonempty}}[s]$, so we get from [NH] that there are $r_3, r_4 \in \mathcal{I}$ with $\mathcal{I} \models [r_3, r_4]\underline{\texttt{head-of}}[s]$. The second case from the definition of $\underline{\texttt{construction}_P}$ applies and we get $\mathcal{I} \models \underline{\texttt{one-step}_P}[r_1, r_2, r_3, r_4]$, this implies $(r_3, r_4) \sqsubset (r_1, r_2) \sqsubset^* (u, u)$. Because of [HSH] there is a $s'' \in \mathcal{I}$ with $\mathcal{I} \models [r_3, r_4, s'']\underline{\texttt{init-of}}[s]$, so we can apply the induction hypothesis to (r_3, r_4). With [CON] and the definition of $\underline{\texttt{one-step}_P}$ the proof is completed. □

Corollary 4.2. *If [INJ], [EPS], [CON], [NOE], [NH], [HSH] and [HS] are fulfilled then*

$$\mathcal{I} \models \underline{\texttt{solvable}_P} \quad \Longrightarrow \quad P \textit{ is solvable}$$

In a first attempt we could require as in Section 4.1 a coding function mapping the set S into $\mathcal{I}$. This will suffice in some examples, but we can be more liberal and allow for each sequence s a "private" coding function for the set of the initial segments of s:

$$\psi \in \prod_{s \in S} (\{0, \ldots, lth(s)\} \to \mathcal{I})$$

The subformulas $\underline{\texttt{nonempty}}$, $\underline{\texttt{init-of}}$ and $\underline{\texttt{head-of}}$ have to work properly for the codings of initial segments:

For all $s \in S, n \leq lth(s)$:

[NIL] $\mathcal{I} \models \underline{\texttt{nonempty}}[\psi(s)(n)]$ iff $n \neq 0$

[HEA] $\mathcal{I} \models [r_1, r_2]\underline{\texttt{head-of}}[\psi(s)(n)]$ iff $n \geq 1$ and $(r_1, r_2) = \phi(s(n))$

[SUB] $\mathcal{I} \models [r_1, r_2, t]\underline{\texttt{init-of}}[\psi(s)(lth(s))]$ iff there is $i \in \{1, \ldots, lth(s)\}$ with $(r_1, r_2) = \phi(s(i))$ and $t = \psi(s)(i-1)$

Lemma 4.4. *If [EPS], [CON], [FIN], [NIL], [HEA] and [SUB] are fulfilled then*

$$P \textit{ is solvable} \quad \Longrightarrow \quad \mathcal{I} \models \underline{\texttt{solvable}_P}$$

Theorem 4.5 gives the complete method developed in this subsection:

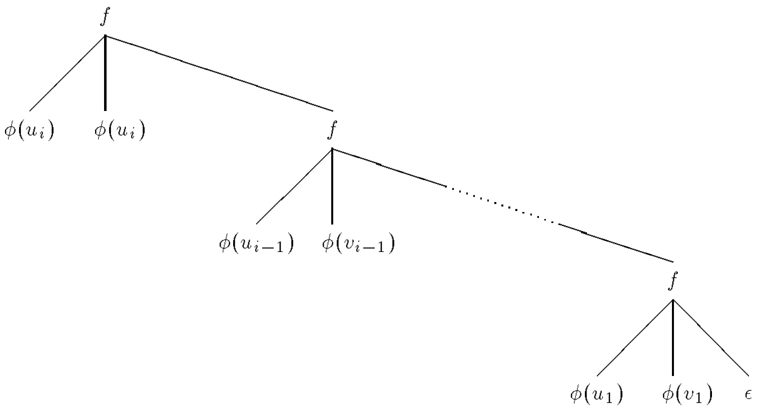

Figure 2. The term $\psi((u_j, v_j)_{j=1\ldots n})(i)$ representing the initial segment $(u_j, v_j)_{j=1\ldots i}$ of the sequence $(u_j, v_j)_{j=1\ldots n}$ for $1 \leq i \leq n$

THEOREM 4.5. *Let B be a predicate logic basis and $\mathcal{I}$ a model for B. If we can find representations ϕ, ψ and formulas $\underline{\texttt{is-}\epsilon}$, $\cdot\underline{\texttt{is}}\cdot\underline{v}$, $\underline{\texttt{finite}}$, $\underline{\texttt{nonempty}}$, $\underline{\texttt{head-of}}$ and $\underline{\texttt{init-of}}$ such that [INJ], [EPS], [CON], [NOE], [NH], [HS], [HSH], [FIN], [NIL], [HEA] and [SUB] are fulfilled, then the first order theory of $\mathcal{I}$ is undecidable.*

EXAMPLE D. Let us now see how the undecidability result for the theory of subterm ordering from Venkataraman (1987) fits into our framework:

Let $F_D := \langle \epsilon(0), a(1), b(1), f(3) \rangle$ and $P_D := \langle =(2), \leq(2) \rangle$. $\mathcal{I}_D$ is the algebra of F_D-ground terms where $\leq$ is interpreted as the subterm relation. The representation of strings has been given in Example (1). We choose $\psi(s)(i)$ as follows (see also Figure 2):

$$\psi(s)(i) := \begin{cases} \epsilon & \text{if } i = 0 \\ f(\phi(u_i), \phi(v_i), \psi(s)(i-1)) & \text{otherwise} \end{cases}$$

and define the remaining formulas:

$$\begin{aligned} (y_1, y_2)\underline{\texttt{head-of}}(x) &:= \exists x'.x = f(y_1, y_2, x') \\ (y_1, y_2, x')\underline{\texttt{init-of}}(x) &:= f(y_1, y_2, x') \leq x \\ \underline{\texttt{nonempty}}(x) &:= \exists y_1, y_2, x'.x = f(y_1, y_2, x') \end{aligned}$$

We can save one alternation of quantifiers in $\underline{\texttt{solvable}}_P$ by transforming $\underline{\texttt{nonempty}}$ into a Π_1 formula†.

$$\underline{\texttt{nonempty}}(x) \quad := \quad x \neq \epsilon \wedge \forall x'.x \neq a(x') \wedge x \neq b(x')$$

THEOREM 4.6. (Venkataraman (1987)) *The (Σ_2-fragment of the) first order theory of the subterm ordering is undecidable.* $\diamond$

† In ground term algebras over a finite alphabet it is always possible to transform a purely equational formula into a Π_1 (or Σ_1) formula, see Comon & Lescanne (1989).

Example E. We can modify the above example by enlarging the model to the algebra of finite and infinite ground terms. We can use exactly the same proof as above but with the $\underline{\texttt{finite}}$ formula as in Example (3) to show

Theorem 4.7. *The (Σ_2 fragment of the) first order theory of the subterm ordering in the algebra of finite and infinite trees in undecidable.*

This undecidability proof for the Σ_2-fragment relies on the finiteness of the particular signature used here. The finiteness of the signature was exploited in Example (3) in order to obtain a sufficiently simple $\underline{\texttt{finite}}$ formula. ◇

Example F. If we drop commutativity from Example (A) we can now show undecidability even of the Σ_2 fragment:

Consider the equational specification $(F_F, A(+))$ where $F_F = \langle \epsilon(0), f(1), +(2)\rangle$ and $A(+)$ denotes the axiom of associativity for $+$:

$$(x+y)+z \quad = \quad x+(y+z)$$

For the initial algebra we take the representation of strings from Example (2) and ψ similar to Example (A):

$$\begin{array}{rcl}
\psi((u_i,v_i)_{i=1,\ldots,m})(j) & := & f(f(\phi(u_j)) + f(f(\phi(v_j)))) + \cdots \\
 & & \quad \cdots + f(f(\phi(u_1)) + f(f(\phi(v_1)))) + \epsilon \quad (j \geq 1) \\
\psi((u_i,v_i)_{i=1,\ldots,m})(0) & := & \epsilon \\
(y_1,y_2)\underline{\texttt{head-of}}(x) & := & \exists x'.x = f(f(y_1) + f(f(y_2))) + x' \\
(y_1,y_2,x')\underline{\texttt{init-of}}(x) & := & x = f(f(y_1) + f(f(y_2))) + x' \vee \\
 & & \quad \exists x''.x = x'' + f(f(y_1) + f(f(y_2))) + x'
\end{array}$$

The definition of the formula $\underline{\texttt{nonempty}}$ is somewhat tedious, since we have to find a Π_1-formula that is equivalent to $\overline{\exists y_1, y_2 (y_1,y_2)}\underline{\texttt{head-of}}(x)$ in $T(F_F)/_{A(+)}$. In the following definition we add in braces as a comment the pattern that x is known to match if the inequalities given so far are fulfilled:

$$\begin{array}{lll}
\multicolumn{2}{l}{\underline{\texttt{nonempty}}(x) := \forall y_1,y_2,y_3,y_4.} & \\
 & x \neq \epsilon \wedge x \neq f(y_1) \wedge x \neq \epsilon + y_1 & \{x \sim f(z_1) + z_2\} \\
\wedge & x \neq f(\epsilon) + y_1 \wedge x \neq f(f(y_1)) + y_2 & \{x \sim f(z_1 + z_2) + z_3\} \\
\wedge & x \neq f(\epsilon + y_1) + y_2 & \{x \sim f(f(z_1) + z_2) + z_3\} \\
\wedge & x \neq f(y_1 + y_2 + y_3) + y_4 \wedge x \neq f(y_1 + \epsilon) & \{x \sim f(f(z_1) + f(z_2)) + z_3\} \\
\wedge & x \neq f(y_1 + f(\epsilon)) \wedge x \neq f(y_1 + f(y_2 + y_3)) + y_4 & \{x \sim f(f(z_1) + f(f(z_2))) + z_3\}
\end{array}$$

Theorem 4.8. *The (Σ_2 fragment of the) theory of a ground term algebra modulo associativity is undecidable for signatures that contain at least one constant, one non-constant function symbol and one associative binary function symbol.* ◇

In the Examples (D) to (F) we gave uniform codings for the sequences. The last Example (G) shows the use of "private" coding functions for the initial segments of a given sequence. As mentioned in the introduction this is an artificial example that serves just for the purpose of demonstrating the usage of our method in its full generality.

Example G. Let $F_G := \langle 0(0), 1(0), +(2), *(2)\rangle$ and $P_G := \langle =(2), \leq(2)\rangle$. Our interpretation $\mathcal{I}_G$ is the model of natural numbers. In order to define the representation of strings we introduce two abbreviations:

$$\begin{aligned} \overline{a}(t) &:= t+t \\ \overline{b}(t) &:= t+t+1 \end{aligned}$$

It is easy to see that the following representation of strings fulfills the requirements since there is an obvious correspondence between strings and the binary representation of natural numbers.

$$\begin{aligned} \phi(\sigma_0\cdots\sigma_n) &:= \overline{\sigma_n}(\cdots\overline{\sigma_0}(1))\cdots) \\ \underline{\texttt{is-}\epsilon}(x) &:= x = 1 \\ (y)\underline{\texttt{is}}(x)\underline{\sigma_0\cdots\sigma_n} &:= y = \overline{\sigma_n}(\cdots\overline{\sigma_0}(x))\cdots) \\ \underline{\texttt{finite}}(x) &:= \textsc{True} \end{aligned}$$

For instance $\phi(aaaba) = 34$, that is 100010 in binary notation. We use Gödel's β-predicate (Gödel (1931)) to represent sequences in the domain of natural numbers. The existence of the representation ψ is a consequence of the fundamental property of the β-predicate. The definition of β and the pertaining theorem are restated in Appendix A.2.

$$\begin{aligned} \underline{\texttt{nonempty}}(c,d,n) &:= n \geq 1 \\ (y_1,y_2)\underline{\texttt{head-of}}(c,d,n) &:= \beta(c,d,n+n,y_1) \wedge \beta(c,d,n+n+1,y_2) \\ (y_1,y_2,(c',d',n'))\underline{\texttt{init-of}}(c,d,n) &:= c' = c \wedge d' = d \wedge n' < n \wedge \\ &\qquad (y_1,y_2)\underline{\texttt{head-of}}(c',d',n'+1) \end{aligned}$$

As a result we obtain the undecidability of the Σ_2 fragment of complete number theory. The reader should note that $\textsc{Im}(\phi) = \mathcal{I}\setminus\{0\}$ — especially the images of ϕ and ψ are not disjoint. ◇

In this section we have finished the presentation of the two methods for proving the undecidability of the first order theory of a model. The first method is appropriate for models that miss a concept of ordering (for instance term algebras modulo associativity and commutativity), while the second is applicable to models where some kind of ordering is present. In view of the fact that the second method can yield undecidability of a simpler fragment than the first one, the question arises why we did not use the second method for proving undecidability of the theory of a partial recursive path ordering in order to find a formula $\underline{\texttt{solvable}_P}$ in a smaller fragment than Σ_4.

The reason is that we can benefit from the simpler quantification structure of the formula $\underline{\texttt{solvable}_P}$ in the second method only if we succeed in finding *simple* formulas $\underline{\texttt{nonempty}}$, $\underline{\texttt{init-of}}$ and $\underline{\texttt{head-of}}$ fulfilling the requirements. More precisely, we get a formula $\underline{\texttt{solvable}_P}$ in Σ_3 iff $\underline{\texttt{nonempty}}$ is in $\Pi_2 \cup \Sigma_1$ and both $\underline{\texttt{init-of}}$ and $\underline{\texttt{head-of}}$ are in $\Pi_1 \cup \Sigma_2$, provided that $\underline{\texttt{is-}\epsilon}$, $\cdot\underline{\texttt{is}}\cdot\underline{v}$ and $\underline{\texttt{finite}}$ do not induce any further alternation of quantifiers. This is usually the case, in all applications we found the expensive operations belong to the datatype set, resp. sequence. Using representations of sequences in the spirit of Example (C) one could try to define $\underline{\texttt{init-of}}$ with the help of a maximality condition as it has been done in the definition of $\underline{\texttt{in}}$, but we did not succeed in finding a formula $\underline{\texttt{init-of}} \in \Pi_1 \cup \Sigma_2$ fulfilling [HSH].

5. Conclusions

We have presented two methods for proving the undecidability of the first order theory of a model. In order to apply one of these methods to a given model we have to find appropriate representations of the data types "string" and "sequence" and formulas expressing the operations on these data types. The two main theorems (Theorem 4.1 and Theorem 4.5) state that the proof of undecidability is completed if the pertaining set of requirements is fulfilled. We would like to point out some statements that at a first glance one might expect to be essential for a reduction proof but that in fact are not. With the presentation of this list we claim that applications benefit from a systematic study of reduction proofs since it localizes the crucial points where the special properties of a model are involved.

1 A general binary concatenation operation is not necessary.
2 The codings of strings and sequences may be non-disjoint.
3 Formulas characterizing the images of the representations ϕ and ψ are not needed. In particular, it is not necessary to express that the elements of some set (sequence) are indeed pairs of strings.
4 One should not worry about an *explicit* characterization of the finiteness of sequences in terms of first order logic.

In the undecidability proof of Venkataraman (1987) there exist subformulas in his construction that *explicitly* specify the shape of the objects that are intended to express P-construction sequences. In the approach presented here this is not necessary, we therefore yield a simpler formula $\underline{\texttt{solvable}}_P$.

There is a potentially useful extension to the method that was not carried out since there are no applications at hand. Strings have been coded in the universe of the model by a representation *function*, instead we could associate an equivalence class of the universe to each string. This implies the need for further restrictions that guarantee the congruence property of the operations on strings.

The starting point of the method proposed is the undecidability of Post's Correspondence Problem. Of course there are many other undecidable problems that might serve for reduction to the decision problem of a theory (see Davis (1977)). One may, for instance, take the uniform halting problem for Turing machines and perform a reduction proof in the above style: A Turing machine halts iff there is a finite sequence of configurations such that the first and the last one are in some special form and such that each adjacent pair is related by some "local transformation". A configuration can be interpreted as a pair of strings (the part of the tape to the left, resp. to the right of the head). Hence the data types involved here are the same, but there is an important advantage of Post's Correspondence Problem that we did exploit in this paper: Pairs of strings are constructed with respect to the Post Correspondence Problem obeying a natural well-founded ordering, while this does not hold for the configurations of a some computation sequence of a Turing machine. So choosing he Halting Problem leads to the difficulty of imposing some well-ordering on the codings of the configurations.

Another popular candidate for reduction is complete number theory. One may use the result of Matijacevič (1970) on the unsolvability of Hilberts Tenth Problem and reduce the Σ_1-fragment hoping to obtain a formula $\underline{\texttt{solvable}}_P$ in a pretty small fragment. In fact Quine (1946) gives a reduction of complete number theory to the theory of concatenation of strings over the alphabet $\{a, b\}$. The number n is coded by the string consisting of n a's, such that addition of numbers corresponds to the concatenation of strings. Multiplication is expressed with the help of lists that can be viewed as computation sequences for an iterative version of the multiplication algorithm. So it seems that this approach yields equally small fragments as ours, but the special point in Quine's proof is that a general concatenation operation is available in the logic, such that no quantifiers are needed for expressing addition. In most of the applications presented here we have just some kind of successor function given, in which case we need a list construction for expressing the addition operation. In order to optimize the alternations of quantifiers the iterative processes of addition and multiplication has to be performed in one list apparatus, thus yielding a more complex reduction proof.

For a similar reason Post's Canonical Production Systems (see Salomaa (1990)) do not qualify for natural candidates . Here the constructibility of a given word is undecidable. As in the cases considered above, we are able to express the notion of a construction sequence easily once we can simulate the elementary construction steps. But in the case of Post's Canonical Production Systems, these elementary construction steps require general string concatenation. For many applications, this will lead to a unnecessary difficulty.

In the introduction we mentioned some decidability results related to our applications, but there are still some gaps between these results and ours. We conclude with a comparison between the undecidability results of this paper and related works on the decidability of special subproblems:

1 The signatures of Theorem 2 are the minimal signatures such that undecidability of the theory of (ground) terms modulo associativity and commutativity holds. Comon (1988) remarks that the case of one AC function symbol plus one constant is equivalent to Presburger arithmetic and is therefore decidable. Using this idea the case of one AC function symbol plus a finite set of constants (called the "theory of finitely generated multisets" in Comon (1991)) can as well be reduced to the theory of Presburger arithmetic. Furthermore the theory of a ground term algebra modulo an empty set of equations has been shown to be decidable in Comon & Lescanne (1989) and Maher (1988).

2 The decidability of the Σ_1-fragment of the theory of ground term algebra modulo associativity and commutativity has been proved in Comon (1988). While we have shown the undecidability of the Σ_3-fragment the Σ_2-case is still unsolved.

3 Comon (1990) shows the decidability of the Σ_1-fragment of a total lexicographic path ordering, but the same question for the partial case remains open. On the other hand we have shown the undecidability of the Σ_4-fragment of the theory of *partial* lexicographic path ordering. We gave the proof for a precedence that is "as total as possible" but did not succeed in applying the technique to the total case. The reason is that at least two incomparable function symbols are needed in order to distinguish between the two components of a pair. A proof of undecidability in the style presented here seems only to be possible beyond a purely symbolic level of representation, as illustrated in Example (G).

I am grateful to Hans-Jürgen Bürckert, Hubert Comon, Jacques Loeckx and Stephan Uhrig for their comments on earlier versions of this paper. The research reported in this article was done while the author was with the "Universität des Saarlandes".

References

Baader, F., Schulz, K. (1991). Unification in the union of disjoint equational theories: Combining decision procedures. Research Report RR–91–33, German Research Center for Artificial Intelligence (DFKI), Stuhlsatzenhausweg 3, W6600 Saarbrücken 11, Germany. To appear at CADE 1992.

Bürckert, H.-J., Schmidt-Schauß, M. (1989). On the solvability of equational problems. SEKI Report SR-89-07, Universität Kaiserslautern, Kaiserslautern, Germany.

Bürckert, H.-J., Herold, A., Schmidt-Schauß, M. (1989). On equational theories, unification and (un)decidability. *Journal of Symbolic Computation*, 7(1,2):3–49.

Comon, H., Lescanne, P. (1989). Equational problems and disunification. *Journal of Symbolic Computation*, 7(3,4):371–425.

Comon, H. (1988). *Unification et Disunification. Théorie et Applications.* PhD thesis, Institut National Polytechnique de Grenoble, Grenoble, France.

Comon, H. (1990). Solving symbolic ordering constraints. *International Journal of Foundations of Computer Science*, 1(4):387–411.

Comon, H. (1991). Disunification: A survey. In Lassez, J.-L., Plotkin, G., editors, *Computational Logic*, chapter 9, pages 322–359. MIT Press.

Courcelle, B. (1983). Fundamental properties of infinite trees. *Theoretical Computer Science*, 25(2):95–169.

Davis, M. (1977). Unsolvable problems. In Barwise, J., editor, *Handbook of Mathematical Logic*, chapter C.2, pages 567–594. North-Holland.

Dershowitz, N., Jouannaud, J.-P. (1990). Rewrite systems. In van Leeuwen, J., editor, *Handbook of Theoretical Computer Science*, volume B, pages 243–320. Elsevier.

Dershowitz, N. (1982). Orderings for term-rewriting systems. *Theoretical Computer Science*, 7:279–301.

Dershowitz, N. (1987). Termination of rewriting. *Journal of Symbolic Computation*, 3:69–116.

Ehrig, H., Mahr, B. (1985). *Fundamentals of Algebraic Specification, vol. 1.* EATCS-Monographs on Theoretical Computer Science. Springer-Verlag.

Enderton, H. B. (1972). *Mathematical Introduction to Logic.* Academic Press.

Fages, F. (1987). Associative-commutative unification. *Journal of Symbolic Computation*, 3(3):257–275.

Feferman, S., editor (1986). *Kurt Gödel, Collected Works.* Oxford University Press. 2 volumes.

Gallier, J. H. (1986). *Logic for Computer Science.* Harper & Row, publishers.

Gödel, K. (1931). Über Formal Unentscheidbare Sätze der Pricipia Mathematica und Verwandter Systeme I. *Monatshefte für Mathematik und Physik*, 38:173–198. Reprinted in Feferman (1986).

Jouannaud, J.-P., Okada, M. (1991). Satisfiability of systems of ordinal notation with the subterm property is decidable. In Albert, J. L., Monien, B., Artalejo, M. R., editors, *18th International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science, vol. 510, pages 455–468, Madrid, Spain. Springer Verlag.

Kirchner, C. (1985). *Méthodes et Outils de Conception Systématique d'Algorithmes d'Unification dans les Théories Équationelles.* PhD thesis, Centre de Récherche en Informatique de Nancy.

Livesey, M., Siekmann, J. (1976). Unification of bags and sets. Technical Report 3-76, Universität Karlsruhe.

Maher, M. J. (1988). Complete axiomatisations of the algebra of finite, rational and infinite trees. In *Proceedings of the Third Annual Symposium on Logic in Computer Science*, pages 348–357. IEEE Computer Society.

Makanin, J. (1977). The problem of solvability of equations in a free semi-group. *Akad. Nauk SSSR*, 232(2).

Matijacevič, Y. (1970). Enumerable sets are diophantine. *Dokl. Akad. Nauk. SSSR*, 191:279–282.

Plotkin, G. D. (1972). Building-in equational theories. In Meltzer, B., Michie, D., editors, *Machine Intelligence 7*, pages 73–90. Edinburgh University Press.

Post, E. L. (1946). A variant of a recursively unsolvable problem. *Bulletin of the AMS*, 52:264–268.

Quine, W. V. (1946). Concatenation as a basis for arithmetic. *Journal of Symbol Logic*, 11(4):105–114.

Rabin, M. O. (1965). A simple method for undecidability proofs and some applications. In Bar-Hillel, Y., editor, *Logic, Methodology and Philosophy of Science*, pages 58–68. North-Holland.

Rogers, Jr., H. (1987). *Theory of Recursive Functions and Effective Computability.* MIT Press, second edition.

Salomaa, A. (1990). Formal languages and power series. In van Leeuwen, J., editor, *Handbook of Theoretical Computer Science*, volume B, pages 103–132. Elsevier.

Siekmann, J. H. (1989). Unification theory. *Journal of Symbol Logic*, 7:207–274.

Stickel, M. E. (1981). A unification algorithm for associative-commutative functions. *Journal of the ACM*, 28(3):423–434.

Tarski, A. (1953). A general method in proofs of undecidability. In *(Tarski* et al. *1953b)*, pages 1–35.

Tarski, A., Mostowski, A., Robinson, R. M. (1953). Undecidability and essential undecidability in arithmetic. In *(Tarski* et al. *1953b)*, pages 37–74.

Tarski, A., Mostowski, A., Robinson, R. M. (1953). *Undecidable Theories.* North-Holland.

Tulipani, S. (1993). Decidability of the existential theory of infinite terms with subterm relation. *Information and Computation*, 103(2). To appear.

Venkataraman, K. N. (1987). Decidability of the purely existential fragment of the theory of term algebra. *Journal of the ACM*, 34(2):492–510.

A. Appendix

A.1. PROOF OF [IN] FOR EXAMPLE (C)

In order to give a proper recursive definition of δ we have to generalize to sequences starting with an arbitrary index $k \geq 1$:

$$\delta((u_i)_{i=k\dots n}) = \begin{cases} \epsilon & \text{if } k > n \\ h(l(a^{k-1}(\epsilon)), e(\phi(u_n)), \delta((u_{i-1})_{i=k+1\dots n})) & \text{otherwise} \end{cases}$$

In order to prove [IN] we need the following lemma:

LEMMA A.1. *Let $s = (u_i)_{i=1\dots n}$ be a nonempty increasing sequence over $\{a,b\}^*$. Then for all $t, t_0 \in T(\{\epsilon, a, b\})$ the following two statements are equivalent:*

1 $\mathcal{I} \models [t]\underline{\mathtt{in}}[\delta(s)]\underline{\mathtt{at}}[t_0]$
2 there exists $j \in \{1, \dots, n\}$ with $t_0 = a^{n-j}(\epsilon)$ and $t = u_j$

PROOF. First we state a simple fact about the lexicographic ordering $\preceq_{\text{lpo}}$ generated by an ordering $\leq_F$ on F:

(*) If $t_1 \preceq_{\text{lpo}} t_2$ then for each operator symbol f occurring in t_1 there is a symbol g in t_2 such that $f \leq_F g$.

Now let $h(l(t_0), e(t), \epsilon) \preceq_{\text{lpo}} \delta((u_i)_{i=k\dots n})$. According to the definition of an lpo there are four possibilities:

1 $h(l(t_0), e(t), \epsilon) \preceq_{\text{lpo}} l(a^{k-1}(\epsilon))$ or $h(l(t_0), e(t), \epsilon) \preceq_{\text{lpo}} e(u_n)$
2 $h(l(t_0), e(t), \epsilon) \preceq_{\text{lpo}} \delta((u_{i-1})_{i=k+1\dots n})$
3 $l(t_0) = l(a^k(\epsilon))$ and $e(t) \preceq_{\text{lpo}} e(u_n)$
4 $l(t_0) \prec_{\text{lpo}} l(a^k(\epsilon))$ and $e(t) \prec_{\text{lpo}} \delta((u_i)_{i=k\dots n})$

Because of (*) possibility (1) can be dropped immediately. For the same reason (4) is only possible if $e(t) \preceq_{\text{lpo}} e(u_j)$ for some $j \in \{k, \dots, n\}$. With an inductive argument in case (2) and applying again (*) we get especially for $k = 1$:

(**) If $h(l(t_0), e(t), \epsilon) \preceq_{\text{lpo}} \delta((u_i)_{i=1\dots n})$ then there are i, i' with $1 \leq i' \leq i \leq n$ such that $t_0 \preceq_{\text{lpo}} a^{n-i}(\epsilon)$ and $t \preceq_{\text{lpo}} u_{i'}$.

<u>**(1) ⇒ (2):**</u> Let $\mathcal{I} \models (t)\underline{\mathtt{in}}(\delta(s))\underline{\mathtt{at}}(t_0)$. From (**) we know that there exist i, i' with

$$\begin{array}{rcl} t_0 & \preceq_{\text{lpo}} & a^{n-i}(\epsilon) \\ t & \preceq_{\text{lpo}} & u_{i'} \end{array}$$

and $1 \leq i' \leq i \leq n$. Since there is no non-constant smaller than a, t_0 must be of the form $a^{n-j}(\epsilon)$ with $i \leq j \leq n$. We therefore conclude

$$t \preceq_{\text{lpo}} u_{i'} \preceq_{\text{lpo}} u_i \preceq_{\text{lpo}} u_j$$

On the other hand we know from the construction of $\delta(s)$ that

$$h(l(a^{n-j}(\epsilon)), e(u_j), \epsilon) \preceq_{\mathrm{lpo}} \delta(s)$$

and (4.4) from the definition of $(\cdot)\underline{\texttt{in}}(\cdot)\underline{\texttt{at}}(\cdot)$ yields $u_j \preceq_{\mathrm{lpo}} t$. From the antisymmetry of $\preceq_{\mathrm{lpo}}$ we get $t = u_j$.

(2) ⇒ (1): (4.3) of the definition of $(\cdot)\underline{\texttt{in}}(\cdot)\underline{\texttt{at}}(\cdot)$ follows immediately from the definition of $\delta(s)$. So let

$$h(l(a^{m-j}(\epsilon)), e(t'), \epsilon) \preceq_{\mathrm{lpo}} \delta(s)$$

Because of (**) there are i, i' with $1 \leq i' \leq i \leq n$ and

$$\begin{array}{rcl} a^{m-j}(\epsilon) & \preceq_{\mathrm{lpo}} & a^{m-i}(\epsilon) \\ t' & \preceq_{\mathrm{lpo}} & u_{i'} \end{array}$$

This is only possible if $j \geq i$ and we obtain

$$t' \preceq_{\mathrm{lpo}} u_{i'} \preceq_{\mathrm{lpo}} u_i \preceq_{\mathrm{lpo}} u_j = t$$

□

A.2. GÖDELS β-PREDICATE

The β predicate was introduced in Gödel (1931). A proof of the theorem we restate below can also be found in textbooks on mathematical logic, for instance Enderton (1972).

$$\beta(x_1, x_2, l, x) := x = x_1 \bmod (1 + (l+1) * x_2)$$

where $x = y \bmod z$ is an abbreviation for

$$\exists q. y = q * z + x \wedge x < z$$

THEOREM A.1. (Gödel (1931)) *For each sequence $a_0, \ldots, a_n$ of natural numbers there exist c, d such that for all $i \leq n$:*

$$\mathcal{I}_G \models \beta[c, d, i, x] \quad \Leftrightarrow \quad x = a_i$$

We can now choose the representation ψ of Example (G):

$$\psi((u_i, v_i)_{i=1,\ldots,m})(j) \quad := \quad (c, d, 2 * j + 1)$$

where c, d are the values corresponding to the sequence

$$(0, 0, \phi(u_1), \phi(v_1), \ldots, \phi(u_m), \phi(v_m))$$