Decision Problems for Term Rewriting systems and Recognizable tree languages[1]

Rémi Gilleron
L.I.F.L, U.R.A 369 C.N.R.S, I.U.T A, Université Lille-Flandres-Artois
59653 Villeneuve d'Ascq Cedex, FRANCE

Abstract. We study the connections between recognizable tree languages and rewrite systems. We investigate some decision problems. Particularly, let us consider the property (P): a rewrite system S is such that, for every recognizable tree language F, the set of S-normal forms of terms in F is recognizable too. We prove that the property (P) is undecidable. We prove that the existential fragment of the theory of ground term algebras modulo a congruence $\overset{*}{\leftrightarrow}_E$ generated by a set E of equations such that there exists a finite, noetherian, confluent rewrite system S satisfying (P) with $\overset{*}{\leftrightarrow}_S = \overset{*}{\leftrightarrow}_E$ is undecidable. Nevertheless, we develop a decision procedure for the validity of linear formulas in a fragment of such a theory.

INTRODUCTION

Equations and rewrite systems have been extensively used to specify programs and data types (see Huet and Oppen [16], Dershowitz and Jouannaud [13] for overviews). The paradigm of Rewrite Systems modelizes evaluation in Logic Programming as well as interpretors in Functional Programming; in this case they usually get the confluence (i.e. the Church-Rosser) property.

The termination and confluence properties are in general undecidable (see [11], [17]). However, if there exists a finite, terminating, confluent rewrite system S such that $\overset{*}{\rightarrow}_S = \overset{*}{\leftrightarrow}_E$, S provides a decision procedure for the equational theory of set of axioms E. Works are devoted to study partial properties or special kinds of rewrite systems. The most popular result is the Knuth-Bendix algorithm, which permits sometimes to compute normal forms of data types. So, researchers get decidable properties of fragments of theories or of subclasses of rewrite systems to supply tools for software engineering.

At the same time, the class of recognizable tree languages defined by finite tree automata is closed under boolean operations, decision algorithms (emptiness, inclusion, equality,...) are known and an important toolbox is available.

---

[1]This work was supported in part by the "PRC Mathématiques et Informatique" and ESPRIT2 Working group ASMICS.

Actually, a signature can be viewed as a bottom-up tree automaton and then the set of terms of sort s is a recognizable tree language (an example is in figure 1). Let us also note that an order-sorted signature is a bottom-up automaton. This last remark is used by H.Comon in [5] in order to introduce "sort constraints" in equational formulas.

In a first part, we study the connections between rewrite systems and recognizable tree languages. Let us give some results and applications in this area.

The set of ground irreducible terms for a left-linear rewrite system is a recognizable tree language ([4],[14]). In this case, finite tree automata theory provides a decision algorithm for ground reducibility of linear terms.

Using tree automata technics, M.Dauchet and S.Tison [12] proved that the theory of ground (i.e. without variables) rewrite systems is decidable (the idea is to encode the rewrite relation in a recognizable tree language).

An important property is the following: a rewrite system S preserves the recognizability if, for every recognizable tree language F, the set S(F) of reductions of terms in F with S is recognizable too.

Indeed, if this property is satisfied, the following different reachability problems for the rewrite system S are decidable and efficient algorithms are available([8]): to decide if t can be reduced in t' with S, to decide if there is some term in a recognizable tree language F (of sort s) which can be reduced in some term of a recognizable tree language F' (of sort s') with S,...

Monadic rewrite systems are an extension of monadic Thue systems, the right-hand sides of monadic systems are terms of depth at most 1. Semi-monadic rewrite systems are such that the right-hand side has the following configuration: $r=f(y_1,...,y_n)$ with f symbol of arity n and y_i is a ground term or a variable. Monadic, semi-monadic and ground rewrite systems preserve recognizability ([20], [9]).

The same problem (to preserve recognizability) for the congruence generated by a set of equations was stated by B.Courcelle in [10] and was shown undecidable in [7].

We investigate in the Section 2 some decision problems for rewrite systems and recognizable tree languages. Particularly, we consider the property (P): a rewrite system S is such that, for every recognizable tree language F, the set of S-normal forms of terms in F is recognizable too. We prove that this property (P) is in general undecidable. We also note some open problems.

In a second part, we study (fragments of) the theory of ground term algebras modulo some congruence $\overset{*}{\leftrightarrow}_E$ generated by a set of equations. Related works are the decidability results of H.Comon in [6] and undecidability results of R.Treinen in [21].

R.Treinen gives a general method for undecidability proofs of first order theories (the idea is to encode the Post correspondence problem in a formula of the theory of a given model). His technique yields the undecidability of the Σ_3-fragment of the theory of ground term algebras in the AC-case and of the Σ_2-fragment in the A-case.

H.Comon proves the decidability of the Σ_1-fragment in the AC-case. Moreover, from the results in [6], one can derive the decidability of the theory of ground term algebras modulo a congruence $\overset{*}{\leftrightarrow}_E$ when E consists in a finite set of ground equations.

We study the same problem for a set E of equations such that there is a finite, noetherian, confluent rewrite system S satisfying (P) and such that $\overset{*}{\leftrightarrow}_S = \overset{*}{\leftrightarrow}_E$. In the Section 3.2, we prove that the existential fragment of the theory of ground term algebras modulo such a congruence is undecidable. Nevertheless, in the Section 3.1, we extend, to the tree case, the results of R.V.Book on decidable sentences of Church-Rosser congruences (see [2]). We obtain a decision algorithm for the validity of a class of linear formulas (and note the undecidability result as soon as we consider non linear formulas). We do not adopt the order-sorted framework but note that the same result is also true if we introduce "sort-constraints" in the formulas as in [5].

1. PRELIMINARIES

1.1. Terms,substitutions

Let Σ be a finite ranked alphabet and T_Σ be the set of terms (trees) over Σ. Let X be a denumerable set of variables, $X_m=\{x_1,\dots,x_m\}$, $T_\Sigma(X)$ and $T_\Sigma(X_m)$ the set of terms over $\Sigma\cup X$ and $\Sigma\cup X_m$. We denote the set of variables occurring in t by *V(t)* . A term t is *linear* if no variable occurs twice in t. A *context* c is a term in $T_\Sigma(X_m)$ such that each variable occurs exactly once in c. The result of the substitution of each x_i by a term t_i is denoted $c(t_1,\dots,t_m)$. We define the set of *occurrences* (or positions) of a term t as usual. Let $t/_o$ denote the subterm of t rooted at occurrence o. We also define a *substitution* as usual. $D(\sigma)$ denotes the domain of a substitution σ and we use the same notation for the substitution and its extension. The trees t and u are *unifiable* if there exists a substitution σ such that $\sigma(t)=\sigma(u)$.

1.2. Rewrite systems

A *rewrite system* $S=\{l \to r \,/\, l, r \in T_\Sigma(X)$, V(r) is included in V(l)$\}$ on T_Σ is a finite set of pairs of terms in $T_\Sigma(X)$ (we only consider finite rewrite systems). $\to_S$ is the rewrite relation induced by S and $\overset{*}{\to}_S$ the reflexive and transitive closure of $\to_S$. We denote the set of irreducible terms for S by *IRR(S)*. Let A be a subset of T_Σ, we denote *the set of reductions of terms in A with S by S(A)* (i.e. $S(A)=\{t' \,/\, \exists t\in A,\ t \overset{*}{\to}_S t'\}$ and *the set of S-normal forms of terms in A by S!(A)* (i.e. $S!(A)=S(A)\cap IRR(S)$).

Note: In the following, we use the notation IRR(S) for the set of ground irreducible terms for S (or the ground S-normal form language).

1.3. Tree languages

A *bottom-up automaton* is a quadruple $A=(\Sigma,Q,Q_f,R)$ where Σ is a finite ranked alphabet, Q is a finite set of states of arity 0, Q_f (set of final states) is a subset of Q, R is a finite set of rules of the following type:$f(q_1,\ldots,q_n)\rightarrow q$ with $n\geq 0$, $q_1,\ldots,q_n$ in Q . Note we can consider R as a ground rewrite system on $T_{\Sigma\cup Q}$, $\rightarrow_A$ is the rewrite relation $\rightarrow_R$.
The tree language recognized by A is $L(A)=\{t\in T_\Sigma \,/\, t \xrightarrow{*}_A q,\ q\in Q_f\}$.
A tree language F is *recognizable* if there exists a bottom-up automaton A such that L(A)=F. The class REC of recognizable tree languages is closed under boolean operations and there exists decision algorithms for emptiness, equality, inclusion of recognizable tree languages.
For more developments see [23].

Figure1

Let the signature S : $0 : \rightarrow Nat.$
$succ : Nat \rightarrow Nat.$
$\Lambda : \rightarrow Stack$
$push : Nat \times Stack \rightarrow Stack$

S can be viewed as the following bottom-up tree automaton
$A=(\Sigma,Q,Q_f,R)$ with $Q = \{q_{Nat}, q_{Stack}\} = Q_f$

$$R = \{\, 0 \rightarrow q_{Nat} \;;\; \begin{array}{c} succ \\ | \\ q_{Nat}\end{array} \rightarrow q_{Nat} \;;\; \Lambda \rightarrow q_{Stack} \;;\; \begin{array}{c} push \\ / \quad \backslash \\ q_{Nat} \quad q_{Stack}\end{array} \rightarrow q_{Stack} \,\}.$$

A term t is of sort Stack iff $t \xrightarrow{*}_A q_{Stack}$

---

2. DECISION PROBLEMS

Problem1:
Instance: A rewrite system S.
Question: For every tree language F in REC, S(F) is in REC.

This problem is in general undecidable.
A proof is in [11] with a confluent and non terminating rewrite system. The problem remains open with for instance a confluent and noetherian rewrite system.

Problem 2:
Instance: A rewrite system S, a tree language F in REC.
Question: S(F) is in REC.

This problem is in general undecidable.
A proof is with the rewrite system defined in [11] and the forest F={k(a,a)}.

Problem 3:
Instance: A rewrite system S.
Question: For every tree language F in REC, S!(F) is in REC.

This problem is in general undecidable.

Sketch of proof:

Let I={1,...,n} and Σ be alphabets. Let ϕ and Ψ be two homomorphisms from I^* in Σ^*. Let $P(\Phi,\Psi)$: ($\exists m \in I^+$, $\Phi(m)=\Psi(m)$) be a Post correspondence problem. We consider, as usual, I and Σ as unary alphabets. Let $\Delta = I \cup \Sigma \cup \{\#, 0, a, \$, \S, A, B, C, D\}$ with #, 0 new symbols of arity 0, a new symbol of arity 1, $ and § new symbols of arity 2, A,B, C and D new symbols of arity 4.

Let R be the rewrite system on T_Δ defined with the metarules given in the figure 2. R is a linear rewrite system.

We now define the rewrite system S.

We add the rules $l \rightarrow 0$ such that the root of l is A, B, C or D and such that, for every term t of root A, B, C or D, either, t matches a left-hand side of a rule in R, or, t matches a left-hand side of one of these new rules. This construction is possible because R is left-linear, we do not detail here this construction.

Moreover, we add to S the rules: for every i in I, $i(0) \rightarrow 0$, for every b in Σ, $b(0) \rightarrow 0$, $a(0) \rightarrow 0$, $\$(x,0) \rightarrow 0$, $\$(0,x) \rightarrow 0$, $\S(0,x) \rightarrow 0$ and $\S(x,0) \rightarrow 0$.

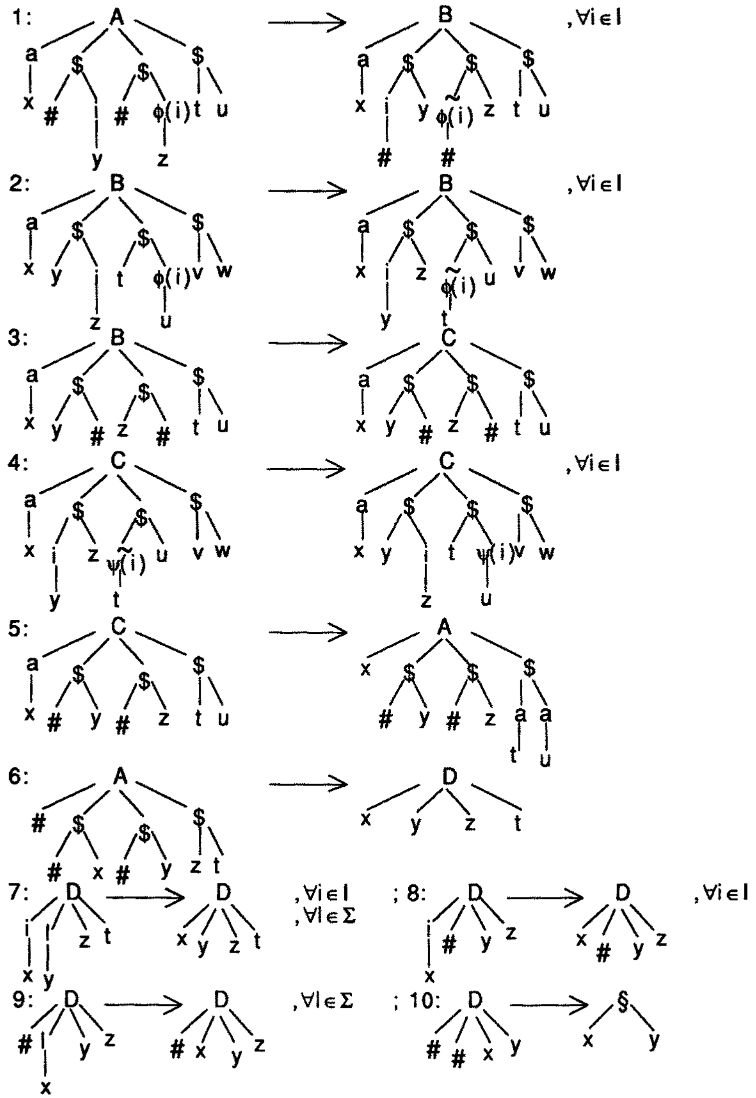
We obtain a linear rewrite system S.

Note: This construction is somewhat complicated because the set of S-normal forms of terms in *every* recognizable tree language F must be recognizable too.

Claim 1: S is terminating (the number of a in the left-most branch is decreasing when we have a cycle).

Claim 2: S is confluent (the critical pairs are S-joinable).

figure2



Let (P): For every recognizable tree language F, the set S!(F) of S-normal forms of terms in F is recognizable too.

Claim 3: (S satisfies (P)) if and only if ($P(\Phi,\Psi)$ has no solution).

Proof of $\Rightarrow$: Let us suppose that $P(\Phi,\Psi)$ has a solution and let us consider the recognizable tree language F defined as follows:
$F=\{A(a^n(\#),\$(\#,u(\#)),\$(\#,v(\#)),\$(\#,\#)) \;/\; u\in I^+,\ v\in \Sigma^*,\ n>0\}$.
Let $t=A(a^n(\#),\$(\#,u(\#)),\$(\#,v(\#)),\$(\#,\#))$ be a term in F.

first case: $v=\Phi(u)$ and $v=\Psi(u)$.
$t \xrightarrow{*}_S t_1=B(a^n(\#),\$(\tilde{u}(\#),\#),\$(\tilde{v}(\#),\#),\$(\#,\#))$ with the rules 1, 2 as $v=\Phi(u)$.
$t_1 \xrightarrow{*}_S t_2=C(a^n(\#),\$(\#,u(\#)),\$(\#,v(\#)),\$(\#,\#))$ with the rules 3, 4 as $v=\Psi(u)$.
$t_2 \to_S t_3=A(a^{n-1}(\#),\$(\#,u(\#)),\$(\#,v(\#)),\$(a(\#),a(\#))$ with the rule 5.
Either $n-1=0$ and we apply the rule 6 or $n-1>0$ and we can reduce t_3 using the rule 1. Thus $t \xrightarrow{*}_S D(u(\#),v(\#),a^n(\#),a^n(\#)) \xrightarrow{*}_S \S(a^n(\#),a^n(\#))$.
second case: $v\neq\Phi(u)$ or $v\neq\Psi(u)$ then $t \xrightarrow{*}_S 0$.Indeed, we reduce t in some term t' in R-normal form which contains one occurrence of A,B or C. With the definition of S, $t' \to_S 0$.
Thus, we can prove that $S!(F)=\{0\}\cup\{\S(a^n(\#),a^n(\#)) \ / \ n>0\}$ and S!(F) is not a recognizable tree language.

Proof of $\Leftarrow$: We suppose: $P(\Phi,\Psi)$ has no solution. Let F be a recognizable tree language, $F=F_0\cup F_{ABCD}\cup F'$ where F_0 is the set of terms in F with at least one occurrence of 0, F_{ABCD} is the set of terms in F with no occurrence of 0 and at least one occurrence of A, B, C or D and F' is the set of terms in F with no occurrence of 0, A, B, C and D. F_0, F_{ABC} and F' are recognizable tree languages. $S!(F_0)=\{0\}$ (with the definition of S), $S!(F')=F'$ (each term in F' is in S-normal form) and then are recognizable tree languages. We have to prove that $S!(F_{ABCD})$ is recognizable.

We consider F such that each term in F contains no occurrence of 0 and at least one occurrence of A, B, C or D. Let t be a term in F.
First case: $t \xrightarrow{*}_R t'$, t' is in R-normal form and t' contains at least one occurrence of A, B, C or D, then, with the construction of S, we have $t \xrightarrow{*}_S t'$ and $t' \xrightarrow{*}_S 0$.
Second case: $t \xrightarrow{*}_R t'$, t' is in R-normal form and t' contains no occurrence of A, B, C and D, then, t' is also in S-normal form. For example, let us consider $t=B(a(\#),\$(\tilde{m}_1(\#),m_2(\#)),\$(\tilde{u}_1(\#),u_2(\#)),\$(t_1,t_2))$ with m_1, m_2 in I^*, u_1, u_2 in Σ^*, t_1 in $I^*(\#)$, t_2 in $\Sigma^*(\#)$. If $\Phi(m_2)=u_2$ and $\Psi(m_1m_2)=u_1u_2$, then $t\xrightarrow{*}_S\S(a(t_1),a(t_2))$. We have to investigate all the possible configurations for such a t. For each case, we define a subset of F which is a recognizable tree language and have to prove that the corresponding set of S-normal forms is recognizable. We omit this technical part of the proof.

As the existence of a solution for a Post correspondence problem is undecidable, we obtain the undecidability of the property (P).

Problem 4:
Instance: A rewrite system S, a tree language F in REC.
Question: S!(F) is in REC.

This problem is in general undecidable.

Proof: We use the previous rewrite system S and the recognizable tree language F defined in the proof of $\Rightarrow$.

Problem 5:
Instance: A rewrite system S.
Question: IRR(S) is in REC.

This problem is open. Note that if S is a left-linear rewrite system, IRR(S) is a recognizable tree language.

Problem 6:
Instance: An homomorphism ϕ and a tree language F in REC.
Question: $\Phi(F)$ is in REC.

This problem is open. Note that if Φ is linear, $\Phi(F)$ is recognizable. We can also note that *if the problem 6 is decidable, then the problem 5 is decidable*. Let $S=\{l_i \to r_i / i \in I,\ l_i, r_i \in T_\Sigma(X)\}$ be a rewrite system on $T_\Sigma(X)$. To each rule of S, we associate a new symbol a_i whose arity is the cardinality of $V(l_i)$. Let $\Delta=\Sigma\cup A$ with $A=\{a_i/i\in I\}$ and let us consider the homomorphism Φ defined with: $\phi(b(x_1,\ldots,x_{r(b)}))=b(x_1,\ldots,x_{r(b)})$ for each b in Σ and $\Phi(a_i(x_1,\ldots,x_{r(a_i)}))=l_i$ for each a_i in A. Let F be the set of terms t in T_Δ such that t contains at least one occurrence of a symbol in A. F is a recognizable tree language. If the problem 6 is decidable, we can decide if $\Phi(F)$ is recognizable, thus we can decide if the complement of $\Phi(F)$ in T_Σ is recognizable and this set is IRR(S).

3. DECIDABLE FRAGMENT OF THEORY

3.1. Decidability in the linear case

Formulas of LinF.

Let Σ be a finite ranked alphabet. Let X be a denumerable set of variables. The set of *terms* is $T_\Sigma(X)$. The *atomic formulas* are $t\equiv t'$ for every terms t and t'. The *logical connectives* are $\neg,\wedge,\vee$. The *quantifiers* are $\exists,\forall$.

We consider the subset *LinF* of formulas defined as follows: the prenex normal form $(Q_1x_1)\ldots(Q_nx_n)\ F(x_1,\ldots,x_n)$ of a formula in LinF satisfies (i), (ii) and (iii) with:

(i): F is linear, i.e. each variable occurs exactly once in F, i.e. every term is linear, no variable occurs twice in an atomic formula and no variable occurs in two atomic formulas.

(ii): For every (non ground) term t in F, in V(t), either, all variables are existentially quantified, or, universally quantified.

(iii): For every atomic formula $t \equiv t'$ occuring in F, all the quantifiers related to variables of V(t) preceed quantifiers related to variables of V(t') or conversely.

Example: $\Sigma=\{0,s,+,^*\}$ with $+,^*$ of arity 2, s of arity 1, 0 of arity 0.
$\forall x\ \exists y\ +(x,s(s(0))) \equiv +(y,s(0))$ is in LinF.
$\forall x\ \exists y\ \exists z\ \ {}^*(x,x) \equiv +({}^*(y,y),{}^*(z,z))$ is not in LinF.

Interpretation

For any given set E of equations on $T_\Sigma(X)$, we define the following interpretation:

For each variable x in X, there is a recognizable tree language D(x) and each variable takes value in its domain D(x).

The symbol $\equiv$ is interpreted as $\overset{*}{\leftrightarrow}_E$.

The quantifiers and connectives are interpreted as usual.

Under this interpretation, each formula in LinF is either true or false as a statement about the congruence $\overset{*}{\leftrightarrow}_E$.

Let *(P): for every recognizable tree language F, the set S!(F) of S-normal forms of terms in F is recognizable too.*

Theorem: *Let E be a set of equations on $T_\Sigma(X)$. If there exists a rewrite system S such that: (*) $\overset{*}{\leftrightarrow}_E = \overset{*}{\leftrightarrow}_S$.*
(**) S is noetherian and confluent.
(***) S satisfies (P)
Then, there is an algorithm that on input a formula in LinF halt and answer whether the formula is true or false under the previous interpretation.

Let us remark that the properties (**) and (***) are in general undecidable. Let us also note that if a rewrite system is such that S preserves the recognizability (the set of reductions of terms in a recognizable tree language is recognizable too) and IRR(S) is recognizable, then the property (***) is ensured with the closure of recognizable tree languages for intersection as $S!(F)=S(F)\cap IRR(S)$.

A monadic rewrite system ([4],[20]) is such that the right-hand side is of depth at most 1 and left-hand side of depth at least 1.

If we consider linear and monadic rewrite systems, we can prove that: termination is decidable, confluence is decidable (using known results of class rewriting [18],[19]), IRR(S) is recognizable (left-linear hypothesis) and recognizability is preserved ([20]). Thus, for linear monadic rewrite systems, the hypothesis are decidable.

Proof: Let $t=t(x_1,\ldots,x_p)$ be a term of $T_\Sigma(X)$. We define D(t) with: if t is in T_Σ, $D(t)=\{t\}$, if $t=t(x_1,\ldots,x_p)$, $D(t)=\{t=t(t_1,\ldots,t_p) \;/\; t_i\in D(x_i)\}$. D(t) is a recognizable tree language, because t is linear. If S satisfies the property (***), then, for each t, S!(D(t)) is also a recognizable tree language.

Given a formula in LinF, with the usual transformations for negations and as the formula is linear, one can distribute the quantifiers over $\wedge$ and $\vee$ so that any formula in LinF is equivalent to a conjunction and disjunction of formulas $(Q_1x_1)\ldots(Q_nx_n)\ f(x_1,\ldots,x_n)$ or $(Q_1x_1)\ldots(Q_nx_n)\ \neg f(x_1,\ldots,x_n)$ satisfying (ii) and (iii) where f is an atomic formula. It is sufficient to investigate these formulas. The different possible cases are:

(1): $t \equiv t'$ with t and t' ground terms.

(2): $\exists\, y_1 \ldots \exists y_p\ \ t(y_1,\ldots,y_p) \equiv t'$ with t' ground term.

(3): $\forall\, x_1 \ldots \forall x_n\ \ t(x_1,\ldots,x_n) \equiv t'$ with t' ground term.

(4): $\exists y_1 \ldots \exists y_p\ \exists y'_1 \ldots \exists y'_{p'}\ \ t(y_1,\ldots,y_p) \equiv t'(y'_1,\ldots,y'_{p'})$.

(5): $\forall\, x_1 \ldots \forall x_n\, \forall x'_1 \ldots \forall x'_{n'}\ \ t(x_1,\ldots,x_n) \equiv t'(x'_1,\ldots,x'_{n'})$.

(6): $\forall\, x_1 \ldots \forall x_n\, \exists y_1,\ldots \exists y_p\ \ t(x_1,\ldots,x_n) \equiv t'(y_1,\ldots,y_p)$.

(7): $\exists y_1 \ldots \exists y_p\, \forall\, x_1 \ldots \forall x_n\ \ t(x_1,\ldots,x_n) \equiv t'(y_1,\ldots,y_p)$.

Case1: It is the word problem for E, obviously decidable reducing t and t' to their S-normal form with the hypothesis (*) and (**).

Case2 and case3: are particular cases of the followings.

Case4: This formula is true under the interpretation if there exists a term u in D(t) and a term v in D(t') such that $u \overset{*}{\leftrightarrow}_E v$; thus, with the hypothesis (*) and (**), if u and v have the same S-normal form. The formula is true under the interpretation if and only if $S!(D(t))\cap S!(D(t'))\neq\emptyset$. The intersection of recognizable tree languages is recognizable, emptiness is decidable for recognizable tree languages.

Case5: This formula is true under the interpretation if S!(D(t)), S!(D(t')) are singleton sets and S!(D(t))=S!(D(t')). These properties are decidable for recognizable tree languages.

Case6: This formula is true under the interpretation if $S!(D(t)\subset S!(D(t'))$ which is also decidable.

Case7: This formula is true under the interpretation if $S!(D(t)\subset S!(D(t'))$ and S!(D(t)) is a singleton set.

We deduce the same result for the formulas $(Q_1x_1)\dots(Q_nx_n)$ $\neg f(x_1,\dots,x_n)$ as they are equivalent to negations of the previous one.❑

Note: We do not adopt the order-sorted framework. The previous proof can be easily modified in this case. Indeed, if we introduce, in our logic, atomic formulas $t \in s$ in order to express the well-formedness, we have to consider the intersection of D(t) with the recognizable tree language of terms of sort s (or his complement for $t \notin s$) in place of D(t).

3.2. The non linear case

Theorem: *Let E be a set of equations on $T_\Sigma(X)$ such that there exists a rewrite system S such that: (*) $\overset{*}{\leftrightarrow}_E = \overset{*}{\leftrightarrow}_S$.*
(**) S is noetherian and confluent.
(***) S satisfies (P)
The existential fragment of the theory of ground term algebras modulo a congruence $\overset{*}{\leftrightarrow}_E$ is undecidable.

Proof: We use the notations of the Section 2. Let $\Delta = I \cup \Sigma \cup \{0,f,\#,\$\}$ with 0,f new symbols of arity 0, # and \$ new symbols of arity 2.
Let $E=\{\#(i(x),\Phi(i)(y))=\#(x,y);\ \$(i(x),\Psi(i)(y))=\$(x,y);\ \#(0,0)=f;\ \$(0,0)=f\}$ be a set of equations on $T_\Delta(X)$. Let S be the rewrite system deduced from E orienting equations from left to right. S is terminating, S is confluent (left-linear and no critical pairs), IRR(S) is a recognizable tree language (S is left-linear), S preserves the recognizability (S is a monadic rewrite system) and obviously $\overset{*}{\leftrightarrow}_E = \overset{*}{\leftrightarrow}_S$.

Consider the formula: $\exists x\ \exists y \bigvee_{i \in I} (\#(i(x),y) \equiv \$(i(x),y))$.

Claim: This formula is true under the interpretation for $\equiv$ if and only if $P(\Phi,\Psi)$.
Proof: The formula is true under the interpretation for $\equiv$ if and only if there exists m in I^+ and u in Σ^* such that $\#(m(0),u(0)) \overset{*}{\leftrightarrow}_E \$(m(0),u(0))$ thus if and only if $\#(m(0),u(0))$ and $\$(m(0),u(0))$ have the same S-normal form. It is possible if and only if $\#(m(0),u(0)) \overset{*}{\rightarrow}_S \#(0,0) \rightarrow_S f$ and $\$(m(0),u(0)) \overset{*}{\rightarrow}_S \$(0,0) \rightarrow_S f$ and then if and only if $u=\Phi(m)=\Psi(m)$.

Note1: If we use the interpretation defined in the Section 3.1, we can use the formula $\exists x\ \exists y\ \#(x,y) \equiv \$(x,y)$ and the domains $D(x)=T_{I\cup\{0\}}-\{0\}$ and $D(y)=T_{\Sigma\cup\{0\}}$.

Note 2: We obtain the same result with: $\Gamma=\Delta\cup\{\S\}$ where § is a new symbol of arity 2, E and S are unchanged , x and y have the same domain and the formula: $\exists x\ \exists y\ \S(\#(x,y),\$(x,y))=\S(f,f)$.❑

This result is related to the result of A.Bockmayr ([1]) on the undecidability of E-unification even in theories represented by canonical term rewriting systems.

CONCLUSION

We apply tree automata technics and results of the tree language theory to rewrite systems. We obtain some decidability results and think that it will be an interesting domain for further research.

Acknowledgments: I sincerely thank M.Dauchet and S.Tison for helpfull discussions and remarks (in particular for the proof of the problem 3). I am also grateful to the referees for pertinent comments on the preliminary version of this paper.

References

[1] A.Bockmayr, A note on a canonical theory with undecidable unification and matching problem, *Journal of Automated Reasoning*, 3, (1987).

[2] R.V. Book, Decidable sentences of Church-Rosser congruences, *Theoretical Computer Science*, 24, (1983).

[3] R.V. Book, Thue systems as Rewriting Systems, *J. of Symbolic Computation*, 3, (1987).

[4] R.V. Book and J.H. Gallier, Reductions in Tree Replacement Systems, *Theoretical Computer Science*, 37, (1985).

[5] H. Comon, Equational formulas in order-sorted algebras, *in Proc. ICALP 90*, (1990).

[6] H.Comon, Unification et disunification: théorie et applications, Thèse de l' I.N.P de Grenoble, France, (1988).

[7] J.L. Coquidé, M.Dauchet and S.Tison, About connections between syntactical and computational complexity, FCT'89, *Lec.Notes.comp.Sci*, 380, (1989).

[8] J.L.Coquidé and R.Gilleron, Proofs and Reachability problems for Rewrite Systems, IMYCS 90, to appear in *Lec.Notes.Comp.Sci*, (1990).

[9] J.L.Coquidé, M.Dauchet, R.Gilleron and S.Vagvölgyi, Tree pushdown automata and Rewrite Systems, submitted paper, (1990).

[10] B. Courcelle,On recognizable sets and tree automata, Resolution of Equations in Algebraic Structures, *Academic Press*, M.Nivat & H. Ait-Kaci edts, (1989).

[11] M.Dauchet, Simulation of Turing machines by a left-linear rewrite-rule, Proc 3rd R.T.A, *Lect. Notes Comput. Sci.* 355, (1989).

[12] M.Dauchet and S. Tison, The theory of Ground Rewrite System is Decidable, *Proc. 5th I.E.E.E symp. on Logic in Computer Science*, (1990).

[13] N.Dershowitz and J.P. Jouannaud, Rewrite systems, Handbook of Theoretical Computer Science, J.V.Leeuwen editor, North-Holland, to appear.(1989).

[14] Z. Fülöp and S. Vàgvölgyi, A characterization of irreducible sets modulo left-linear rewriting systems by tree automata, *Fundamenta Informaticae XIII*, (1990).

[15] F. Gecseg and M. Steinby, Tree automata, *Akademiai Kiado*, (1984).

[16] G.Huet and D.C. Oppen, Equations and Rewrite Rules: A survey, in R.V.Book, ed., New York, *Academic Press*, Formal Language Theory: Perspectives and Open Problems, (1980).

[17] M.Jantzen, Confluent string rewriting, EATCS Monographs on Theoretical Computer Science 14, *Springer Verlag*, (1988).

[18] J.P.Jouannaud and H.Kirchner, Completion of a set of rules modulo a set of equations, *SIAM J. Comput.* 15, (1986).

[19] G.Peterson and M.Stickel, Complete sets of reductions for some equational theories, *J. Assoc. Comput. Mach. 28*, (1981).

[20] K. Salomaa, Deterministic Tree Pushdown Automata and Monadic Tree Rewriting Systems, *Journal of Comput. and Syst. Sci.*, 37, (1988).

[21] R.Treinen, A new method for undecidability proofs of first order theories, T.R. A-09/90, Universität des Saarladandes, Saarbrücken, (1990).