The Word Problem of ACD-Ground theories is Undecidable

Claude Marché
Laboratoire de Recherche en Informatique (UA 410 CNRS)*
Bat 490, Université Paris-Sud
91405 ORSAY CEDEX, FRANCE
E-mail: marche@lri.lri.fr

Abstract

We prove that there exists an ACD-ground theory — an equational theory defined by a set of ground equations plus the associativity and commutativity of two binary symbols $*$ and $+$, and the distributivity of $*$ over $+$ — for which the word problem is undecidable.

1 Introduction

Equations are ubiquitous in mathematics and the sciences.

The word problem of a given a set of equations (that is the problem of deciding if an identity is a consequence of the equations), or equivalently of its equational theory, is undecidable in general. But there are known classes of equational theories which have a decidable word problem, in particular, *ground* equational theories.

The most famous examples of theories with undecidable word problem are given by sets of ground equations over word algebras. Such theories can be considered as *associative*-ground theories over a certain term algebra, whose signature contains only constants besides the binary (associative) symbol. Their word problem is known to be undecidable in general since 1947 [11, 14]. One can find an explicit example in [12].

Completion procedures originally introduced in [9], are a very general way to decide the word problem of a set of equations. But since such algorithms do not run in finite time in general, they yield only semi-decision procedures. Consequently, if one wants to decide the word problem for a given class of equational theories by a completion method, one must prove that the procedure terminates for all inputs. Using this method, it has been recently proved [13] (see also [10]) that *associative-commutative*-ground theories, given by a set of ground axioms and arbitrarily many associative-commutative operators, all have a decidable word problem. In fact, this last result is much stronger because it says that the uniform AC-ground word problem is decidable, and more precisely that every AC-ground theory has a canonical rewrite system, obtained in finite time by an AC-ground completion procedure.

In this paper, we consider *associative-commutative-distributive*-ground theories, given by ground equations and two AC-operators together with the distributivity law. We prove that there exists a ACD-ground theory with an undecidable word problem.

The way we prove this result is similar to the proof of undecidability of associative-ground theories: we encode the halting problem of a Turing machine, known to be undecidable since 1936 [15, 16].

We start in section 2 with a few definitions and notations. In section 3, we recall the main results we use on Turing machines. The proof of our result is given in section 4.

*This work is partly supported by the "GRECO de programmation du CNRS" and Basic Research Action COMPASS

2 Definitions and notations

We assume known the standard definitions and notations about rewriting techniques, given in [6]. We denote by $\mathbb{N}$ the set of natural numbers and by $\mathbb{Z}$ the set of integers.

Definition 2.1 *The theory* $\mathrm{AC}(+)$ *for a binary function symbol* $+$ *is the one given by*

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

The theory $\mathrm{ACD}(*,+)$ *for two binary function symbols* $*$ *and* $+$ *is defined by* $\mathrm{AC}(*)$, $\mathrm{AC}(+)$ *and*

$$x*(y+z)=x*y+x*z$$

Definition 2.2 *For a signature* $\mathcal{F}$ *which contains two binary symbols* $*$ *and* $+$*, The ACD-ground theory defined by a set* E *of ground equations is the equational theory of* $E \cup \mathrm{ACD}(*,+)$.

Definition 2.3 *The ACD-ground word problem of a set of ground equations* E *is:*
Instance: *two terms s,t in* $T(\mathcal{F})$.
Question: *does* $s =_{E\cup \mathrm{ACD}} t$ *?*

3 Turing machines

Our definition of Turing machines is taken from [2]: a Turing machine is deterministic, has an infinite tape in both directions, and the characters of the tape belong to $\{0,1\}$.

Definition 3.1 *A* Turing machine *is a triple* $\langle Q, q_0, \delta\rangle$ *where* Q *is a finite set (the set of states),* $q_0 \in Q$ *(the initial state), and* δ *(the transition function) is a partial function from* $Q\times\{0,1\}$ *to* $Q\times\{0,1\}\times\{L,R\}$.

Definition 3.2 *A* configuration *of a Turing machine* $\mathcal{M}=\langle Q,q_0,\delta\rangle$ *is a triple* $\langle T,p,q\rangle$ *where* T *(the tape) is a mapping* $\mathbb{Z}\mapsto\{0,1\}$, $p\in\mathbb{Z}$ *(the current position) and* $q\in Q$ *(the current state).*

We also write $T=\ldots\alpha_{-1}\alpha_0\alpha_1\alpha_2\ldots$ if $T(p)=\alpha_p$.

Definition 3.3 $\mathcal{M}$ transforms *a configuration* $\langle T,p,q\rangle$ *into another* $\langle T',p',q'\rangle$ *if and only if* $\delta(q,T(p))=(q',\alpha,d)$, $T'(p)=\alpha$, $\forall x\neq p\ T'(x)=T(x)$*, and* $p'=p-1$ *if* $d=L$, $p'=p+1$ *if* $d=R$.

We denote this as $\langle T,p,q\rangle \underset{\mathcal{M}}{\Rightarrow} \langle T',p',q'\rangle$,

Intuitively, $\delta(q,\alpha)=(q',\beta,d)$ means "if $\mathcal{M}$ is in state q and α is the symbol at the current position on the tape, then β is written on the tape, the state becomes q', and the current position is moved towards the direction indicated by d, L for left and R for right. If $\delta(q,\alpha)$ is undefined, then the machine halts.

Definition 3.4 *The* computation *of* $\mathcal{M}$ *on the initial data* $\alpha_1\ldots\alpha_n$ *is the possibly infinite sequence of configurations obtained from the initial one* $\langle T_0,0,q_0\rangle$ *where* $T_0=\ldots00\alpha_1\ldots\alpha_n00\ldots$, $T_0(0)=\alpha_1$.

$\mathcal{M}$ halts *if the sequence is finite.*

Definition 3.5 *The halting problem of a Turing machine $\mathcal{M}$ is:*
Instance: *an initial data $\alpha_1 \ldots \alpha_n$.*
Question: *does the computation of $\mathcal{M}$ on $\alpha_1 \ldots \alpha_n$ terminate ?*

For any natural number $n \in \mathbb{N}$, we denote by $\overline{n}$ the word $\underbrace{1 \ldots 1}_{n+1}$.

Definition 3.6 *A k-ary partial function $\mathbb{N}^k \mapsto \mathbb{N}$ is said to be* recursive *if there exists a Turing machine $\mathcal{M}$ such that if we start $\mathcal{M}$ on the initial data $\overline{x_1}0\overline{x_2}0 \ldots 0\overline{x_k}$ then*

- *if $f(x_1, \ldots, x_k)$ is defined then $\mathcal{M}$ halts scanning the leftmost symbol of $\overline{f(x_1, \ldots, x_k)}$.*
- *if $f(x_1, \ldots, x_k)$ is not defined then $\mathcal{M}$ never halts.*

Definition 3.7 *A set of k-uples $E \subset \mathbb{N}^k$ is said to be recursive if its characteristic function $\chi_E : \mathbb{N}^k \mapsto \{0, 1\}$ is recursive.*

The following result proves that there exists a Turing machine with an undecidable halting problem:

Theorem 3.1 *There exists a recursive function f such that $Dom(f)$ is not recursive.*

This theorem means that there is no algorithm for deciding if the Turing machine of f halts for a given input.

4 Undecidability of the ACD-ground word problem

Given a Turing machine, we are going to encode its halting problem into an ACD-ground word problem.

Our method is similar to the proof of undecidability of the A-ground word problem [5]. In this proof, a configuration of the machine is coded by a word on a certain alphabet. Here we don't have any associative operator so we can't use words.

The important property of words is that they are *ordered* sequences of letters. Here we are going to use *polynomials*, where the coefficients are letters, in order to code ordered sequences. Polynomials are easy to encode by two AC-operators with distributivity.

4.1 Representation of a configuration by a polynomial

Let $\mathcal{M} = \langle Q, q_0, \delta \rangle$ be an arbitrary Turing machine. Let $Q = \{q_0, \ldots, q_N\}$. We start first by giving the alphabet for coding $\mathcal{M}$:

Definition 4.1 *Let $Q' = \{q'_0, \ldots, q'_N\}$. The signature $\mathcal{F}(\mathcal{M})$ (or $\mathcal{F}$ for short) is*

$$Q \cup Q' \cup \{q, q', q'', h, h', X, I, U, 0, 1\} \cup \{*, +\}$$

where every symbol other than $*$ and $+$ are constants.

Definition 4.2 *A configuration $\langle T, p, q_j \rangle$ where $T = \ldots 00\alpha_1 \ldots \alpha_n 00 \ldots$ and $T(p) = \alpha_i$ for some i is coded by the following term (a polynomial in X):*

$$\overline{\langle T, p, q_j \rangle} = h + X\alpha_1 + \ldots + X^{i-1}\alpha_{i-1} + X^i q_j + X^{i+1}\alpha_i + \ldots + X^{n+1}\alpha_n + X^{n+2}h'$$

q and q' will simulate an halting state, h and h' are the "bounds" of the tape (they represents infinitely many 0). I and U will allow us to add a 0 on the left of the tape if needed.

We have to ensure that there is always a 0 or a 1 to read on the right of the character q_j, that is if i is the exponent of the monomial whose coefficient is q_j, then there is one (and only one !) monomial $X^{i+1}0$ or $X^{i+1}1$. That means that if we go too far to the left or to the right, we must add a 0 to the tape. Adding a 0 on the right is trivial because only the exponent of h' have to be changed. But when adding a 0 on the left, we have to increment by 1 every exponent. We will describe how this is done in the next section.

4.2 Simulating the behaviour of the machine by a set of rewrite rules

The behaviour of the Turing machine is simulated by rewriting modulo ACD with a set of ground rules uniquely defined by the machine.

Definition 4.3 *$\mathcal{M}$ defines a set of ground rules $R(\mathcal{M})$ as follows:*

- *for each transition $\delta(q_i, \alpha) = (q_j, \beta, R)$, $R(\mathcal{M})$ contains*

$$\begin{aligned} q_i + X\alpha + X^2 0 &\rightarrow \beta + Xq_j + X^2 0 && \text{(a)} \\ q_i + X\alpha + X^2 1 &\rightarrow \beta + Xq_j + X^2 1 && \text{(b)} \\ q_i + X\alpha + X^2 h' &\rightarrow \beta + Xq_j + X^2 0 + X^3 h' && \text{(c)} \end{aligned}$$

- *for each transition $\delta(q_i, \alpha) = (q_j, \beta, L)$, $R(\mathcal{M})$ contains*

$$\begin{aligned} 0 + Xq_i + X^2\alpha &\rightarrow q_j + X0 + X^2\beta && \text{(d)} \\ 1 + Xq_i + X^2\alpha &\rightarrow q_j + X1 + X^2\beta && \text{(e)} \\ h + Xq_i + X^2\alpha &\rightarrow h + Xq'_j + X^2 0 + X^3 I\beta && \text{(f)} \end{aligned}$$

- *for each undefined $\delta(q_i, \alpha)$,*

$$q_i + X\alpha \rightarrow q + X\alpha \qquad \text{(g)}$$

- *for every $\alpha, \beta \in \{0, 1\}$, $R(\mathcal{M})$ contains:*

$$\begin{aligned} I\alpha + \beta &\rightarrow \alpha + XI\beta && \text{(h)} \\ I\alpha + h' &\rightarrow U\alpha + Xh' && \text{(i)} \\ \alpha + UX\beta &\rightarrow U\alpha + X\beta && \text{(j)} \\ q'_j + UX\beta &\rightarrow q_j + X\beta && \text{(k)} \end{aligned}$$

These rules perform the operation of adding a 0 to the left of the tape.

- *for every* $\alpha \in \{0,1\}$*:*

$$\begin{array}{rcl} q + X\alpha & \to & \alpha + Xq \qquad \text{(l)} \\ q + Xh' & \to & q' \qquad \text{(m)} \\ \alpha + Xq' & \to & q' \qquad \text{(n)} \\ h + Xq' & \to & q'' \qquad \text{(o)} \end{array}$$

These rules rewrite every term which represent a halting state to q''*.*

We are also interested in the equational theory defined by these rules so we define the set of equations defined by $\mathcal{M}$:

Definition 4.4 *The set* $E(\mathcal{M})$ *of ground equations is* $\{s = t \mid s \to t \in R(\mathcal{M})\}$*.*

Example We give now an example to illustrate the role of rules (h)-(k). Assume that the tape is $\ldots 001100 \ldots$, the current position on the leftmost 1, and the current state is q_0. A term that represents this configuration is:

$$h + Xq_0 + X^2 1 + X^3 1 + X^4 h'$$

Now let us assume there is transition $\delta(q_0, 1) = (q_1, 0, L)$. $R(\mathcal{M})$ contains then the rule:

$$h + Xq_0 + X^2 1 \to h + Xq_1' + X^2 0 + X^3 I0$$

So our term rewrites to

$$h + Xq_1' + X^2 0 + X^3 I0 + X^3 1 + X^4 h'$$

Now the state has a prime so every transition is blocked. Only rule (h) can be applied, modulo ACD:

$$X^3 I0 + X^3 1 =_{\mathrm{ACD}} X^3(I0 + 1) \underset{(h)}{\longrightarrow} X^3(0 + XI1) =_{\mathrm{ACD}} X^3 0 + X^4 I1$$

we obtain then this sequence of reductions:

$$\begin{array}{rl} h + Xq_1' + X^2 0 + X^3 I0 + X^3 1 + X^4 h' & \xrightarrow[(h)/\mathrm{ACD}]{} \\ h + Xq_1' + X^2 0 + X^3 0 + X^4 I1 + X^4 h' & \xrightarrow[(i)/\mathrm{ACD}]{} \\ h + Xq_1' + X^2 0 + X^3 0 + X^4 U1 + X^5 h' & \xrightarrow[(j)/\mathrm{ACD}]{} \\ h + Xq_1' + X^2 0 + X^3 U0 + X^4 1 + X^5 h' & \xrightarrow[(j)/\mathrm{ACD}]{} \\ h + Xq_1' + X^2 U0 + X^3 0 + X^4 1 + X^5 h' & \xrightarrow[(k)/\mathrm{ACD}]{} \\ h + Xq_1 + X^2 0 + X^3 0 + X^4 1 + X^5 h' & \end{array}$$

and we see that 1 has been added to each exponent. Now the rule corresponding to the transition $\delta(q_1, 0)$ can apply.

4.3 Proof of the correctness of the simulation

Our goal is to prove that rewriting with $R(\mathcal{M})$ modulo ACD simulates exactly the behaviour of $\mathcal{M}$. The main point is to show that $R(\mathcal{M})$ is deterministic. This not trivial because there are overlaps between rules of $R(\mathcal{M})$.

We start with a few preliminary lemmas:

Lemma 4.1 *If two terms are equal modulo $E(\mathcal{M}) \cup$ ACD then they have the same number of characters belonging to $\tilde{Q} = \{q_0, \ldots, q_N, q'_0, \ldots, q'_N, q, q'\}$.*

Proof. All equations keep the number of elements of $\tilde{Q}$ the same. □

In the following, we want to consider terms which have a meaning for us, that is they represent a valid state of the machine. First of all, we consider terms u which have exactly one symbol in $\tilde{Q}$. This symbol is called the *state* of u. We then define a set of terms which are the valid terms.

Definition 4.5 *The set of* valid terms *is the union of the following six sets. The first one is the one which contains terms representing a configuration of the machine. The five others, represents transitory configurations when we perform the special operations of adding a 0 on the left of the tape and deleting everything if we have gone to an halting state:*

- *"normal" configuration (the state of u is in $\{q_0, \ldots, q_N\}$ and u does not contain any I or U):*

$$S_1 = \{h + X\alpha_1 + \ldots + X^{i-1}\alpha_{i-1} + X^i q_j + X^{i+1}\alpha_{i+1} + \ldots + X^{n+1}\alpha_n + X^{n+2}h'\}$$

- *adding a 0 on the left, first step, the state of u is in $\{q'_0, \ldots, q'_N\}$ and u contains exactly one I and no U, and is of the form:*

$$S_2 = \{h + Xq'_j + X^2\alpha_1 + \ldots + X^i\alpha_{i-1} + X^{i+1}I\alpha_i + X^{i+1}\alpha_{i+1} + \ldots + X^n\alpha_n + X^{n+1}h'\}$$

- *adding a 0 on the left, second step, the state of u is in $\{q'_0, \ldots, q'_N\}$ and u contains exactly one U and no I, and is of the form:*

$$S_3 = \{h + Xq'_j + X^2\alpha_1 + \ldots + X^i\alpha_{i-1} + X^{i+1}U\alpha_i + X^{i+2}\alpha_{i+1} + \ldots + X^{n+1}\alpha_n + X^{n+2}h'\}$$

- *the machine stops, and we want to reduce to q''. We first want to move the current position completely to the right. the state of u is q and u does not contain any I or U, and is of the form:*

$$S_4 = \{h + X\alpha_1 + \ldots + X^{i-1}\alpha_{i-1} + X^i q + X^{i+1}\alpha_{i+1} + \ldots + X^{n+1}\alpha_n + X^{n+2}h'\}$$

- *we remove every symbols 0 and 1 form the tape. the state of u is q' and u does not contain any I or U, and is of the form:*

$$S_5 = \{h + X\alpha_1 + \ldots + X^n\alpha_n + X^{n+1}q'\}$$

- *final term, representing the final configuration.*

$$S_6 = \{q''\}$$

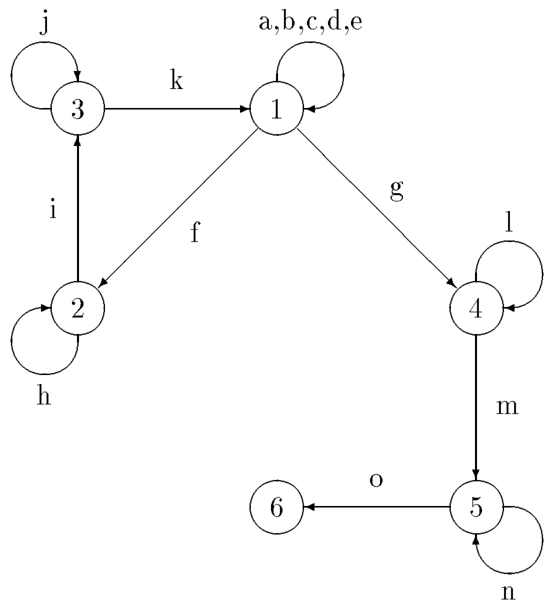

Figure 1: Source and Destination of rules of $R(\mathcal{M})$

This is straightforward that the S_i are disjoints by looking at the state and the presence of I or U:

Lemma 4.2 *the sets S_i are disjoints.*

We prove now that each rule of $R(\mathcal{M})$ can be applied only to terms which belong to a particular set S_i and always produces a term in a particular S_j. These sets will be called the source and the destination of the considered rule.

Definition 4.6 *For each rule (r) in $R(\mathcal{M})$ the source and the destination of this rule, denoted* $\mathrm{Source}(r)$ *and* $\mathrm{Dest}(r)$ *is one of the S_i as shown by the diagram of figure 1.*

For example, $\mathrm{Source}(g) = S_1$ and $\mathrm{Dest}(g) = S_4$.

The following lemma shows that rules of $R(\mathcal{M})$ always produce a valid term from a valid term, and conversely lead to a valid term only from valid terms.

Lemma 4.3 *Assume $s \xrightarrow[(r)/\mathrm{ACD}]{} t$ for arbitrary terms s, t and rule (r) in $R(\mathcal{M})$. The following properties are equivalent:*

(i) *s is valid.*

(ii) $s \in \mathrm{Source}(r)$

(iii) $t \in \mathrm{Dest}(r)$.

(iv) *t is valid.*

Proof. (ii)$\Longrightarrow$(i) and (iii)$\Longrightarrow$(iv) are trivial. Converse implications and (ii)$\Longleftrightarrow$(iii) are obvious and left to the reader. $\square$

Lemma 4.4 *Let u be any valid term. There is at most one rule that can be applied to u, at only one position in u.*

Proof. by lemma 4.3 and because there is no overlap between rules which have the same source. □

We obtain now the following theorem which shows that $R(\mathcal{M})/\mathrm{ACD}$ simulates the behaviour of $\mathcal{M}$.

Theorem 4.1 $\langle T, p, q\rangle \overset{*}{\underset{\mathcal{M}}{\Rightarrow}} \langle T', p', q'\rangle$ *if and only if* $\overline{\langle T, p, q\rangle} \xrightarrow[R(\mathcal{M})/\mathrm{ACD}]{*} \overline{\langle T', p', q'\rangle}$

Corollary 4.1 *Given the input* $\alpha_1 \ldots \alpha_n$*, a Turing machine* $\mathcal{M}$ *stops if and only if*

$$h + Xq_0 + X^2\alpha_1 + \ldots + X^{n+1}\alpha_n + X^{n+2}h' \xrightarrow[R(\mathcal{M})/\mathrm{ACD}]{*} q''$$

Proof. Straightforward by lemma 4.4. □

This shows in particular that the problem of *reducibility* modulo $R(\mathcal{M})/\mathrm{ACD}$ is undecidable.

4.4 Reduction of the halting problem into an ACD-ground word problem

We have shown in the previous section that the halting problem of $\mathcal{M}$ reduces to the reducibility problem of $R(\mathcal{M})/\mathrm{ACD}$. But in general, the reducibility problem of a rewrite system R is not equivalent to the word problem of its equational theory. The following theorem proves that we have this equivalence for $R(\mathcal{M})$ if valid terms only are considered.

Theorem 4.2 *For a given machine* $\mathcal{M}$ *and any data* $\alpha_1 \ldots \alpha_n$ *we have*

$$h + Xq_0 + X^2\alpha_1 + \ldots + X^{n+1}\alpha_n + X^{n+2}h' \xrightarrow[R(\mathcal{M})/\mathrm{ACD}]{*} q''$$

if and only if

$$h + Xq_0 + X^2\alpha_1 + \ldots + X^{n+1}\alpha_n + X^{n+2}h' \xleftrightarrow[E(\mathcal{M})\cup\mathrm{ACD}]{*} q''$$

Proof. One direction is obvious. For the other one, let us consider a proof

$$h + Xq_0 + X^2\alpha_1 + \ldots + X^{n+1}\alpha_n + X^{n+2}h' = u_1 \xleftrightarrow[R(\mathcal{M})]{} u_2 \xleftrightarrow[R(\mathcal{M})]{} \ldots \xleftrightarrow[R(\mathcal{M})]{} u_n = q''$$

($\xleftrightarrow[R(\mathcal{M})]{}$ means that the rewriting can be done from left to right as well as from right to left).

It is obvious from lemma 4.3 that for each elementary step $u_i \xrightarrow[(r)]{} u_{i+1}$, $u_i \in \mathrm{Source}(r)$ and $u_{i+1} \in \mathrm{Dest}(r)$; and for each $u_i \xleftarrow[(r)]{} u_{i+1}$, $u_i \in \mathrm{Dest}(r)$ and $u_{i+1} \in \mathrm{Source}(r)$.

Assume this proof has a "peak" $u_{i-1} \leftarrow u_i \rightarrow u_{i+1}$. By lemma 4.4 we know that only one rule can be applied to u_i, and only at a given position, hence u_{i-1} and u_{i+1} are identical. The proof can then be made shorter by removing the two steps involved in the peak. Repeating this process, we obtain a proof without peak. But q'' is in normal form hence we have in fact a rewrite proof from left to right. □

We obtain then from theorem 3.1:

Theorem 4.3 *There is a set* E *of ground axioms on a vocabulary* $\mathcal{F} \cup \{*, +\}$ *such that the word problem for* $E \cup \mathrm{ACD}(*, +)$ *is undecidable.*

5 Conclusion

Table 1 on page 10 shows known results on decidability or undecidability of word problem of E-ground theories for various set of axioms E. In the cases where the word problem is decidable, this is a consequence of the termination of E-ground completion, so the result is much stronger: every E-ground theory has a canonical rewrite system.

We see that the status of the E-ground word problem (decidable or undecidable) is very sensitive to the addition of new axioms to E.

We can also mention Buchberger's algorithm for computing Gröbner basis of polynomial ideals, which is also a kind of ground completion procedure, and is known to terminate for any input [3]. It is a decision procedure for the word problem of the theory of the considered polynomial ring $R[X_1, \ldots, X_n]$. Notice that in Buchberger's algorithm, R must be a field, so in general it is not an equational theory, but there are extensions [4, 7, 8] which allow R to be, for example, any Euclidean ring. In particular, it is possible to compute Gröbner basis for polynomials with integral coefficients, hence this is a decision procedure for the word problem of an E-ground theory where E is the (multi-sorted) equational theory of polynomial rings.

Further work will be to find abstract properties on the theory E which make the E-ground word problem decidable or not. We will try to investigate other examples for E.

Bibliography

1. Leo Bachmair and David A. Plaisted. Termination orderings for associative-commutative rewriting systems. *Journal of Symbolic Computation*, 1(4):329–349, December 1985.

2. Jon Barwise, editor. *Handbook of Mathematical Logic*, chapter Recursion Theory. North-Holland, 1977.

3. Bruno Buchberger. *An Algorithm for Finding a Basis for the Residue Class Ring of a Zero-Dimensional Ideal.* PhD thesis, University of Innsbruck, Austria, 1965. (in German).

4. Reinhard Bündgen. Simulating Buchberger's algorithm by a Knuth-Bendix completion procedure. In Ronald V. Book, editor, *Proc. 4th Rewriting Techniques and Applications, Como, LNCS 488.* Springer-Verlag, April 1991.

5. Martin Davis. Unsolvable problems. In J. Barwise, editor, *Handbook of Mathematical Logic*, pages 567–594. North-Holland, 1977.

6. Nachum Dershowitz and Jean-Pierre Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 243–309. North-Holland, 1990.

7. Abdelilah Kandri-Rody and Deepak Kapur. An algorithm for computing the Gröbner basis of a polynomial ideal over an Euclidean ring. Technical Report 84CRD045, CRD, General Electric Company, Schenectady, New-York, December 1984.

8. Deepak Kapur and Paliath Narendran. Constructing Gröbner bases for a polynomial ideal. Presented at the workshop *Combinatorial algorithms in algebraic structures* at Europaische Akademie, Otzenhause, 1985.

9. Donald E. Knuth and Peter B. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, 1970.

E	E-compatible ordering total on ground terms	E-ground completion algorithm	Word problem of E-ground theories
C [1]	exists [2]	terminates	decidable
A [3]	exists [4]	does not terminate	undecidable [5]
AC [6]	exists [7]	terminates [8]	decidable
ACD [9]	exists [10]	does not terminate	undecidable [11]
$R[X_1, \ldots, X_n]$ [12]	(not needed)	terminates [13]	decidable

Notes:

1. With arbitrarily many commutative operators (in particular 0, that is E is the empty theory).
2. Recursive Path Ordering, total precedence, multiset status for commutative operators, lexicographic status for others.
3. With one associative operator.
4. Recursive Path Ordering, total precedence, lexicographic status, flattened terms, . minimal in the precedence.
5. Proved independently by Post and Markov in 1947 [11, 14].
6. With arbitrarily many associative-commutative operators.
7. Narendran and Rusinowitch, 1990 [13].
8. Proved independently by Narendran and Rusinowitch and by Marché in 1990 [10, 13].
9. With two AC-operators, the one distributing over the other.
10. Associative Path Ordering, Bachmair and Plaisted, 1985 [1].
11. The result of this paper.
12. Polynomials over a ring R.
13. Buchberger, 1965 [3]. See text for more details.

Table 1: Status of the word problem for E-ground theories for various E.

10. Claude Marché. On ground AC-completion. In Ronald V. Book, editor, *Proc. 4th Rewriting Techniques and Applications, Como, LNCS 488*. Springer-Verlag, April 1991.

11. A. A. Markov. On the impossibility of certain algorithms in the theory of associative systems. *Dokl. Akad. Nauk SSSR*, 55(7):587–590, 1947. In Russian, English translation in C.R. Acad. Sci. URSS, 55, 533-586.

12. J. V. Matijasevic. Simple examples of undecidable associative calculi. *Soviet Mathematics (Dokladi)*, 8(2):555–557, 1967.

13. Paliath Narendran and Michaël Rusinowitch. Any ground associative-commutative theory has a finite canonical system. In Ronald V. Book, editor, *Proc. 4th Rewriting Techniques and Applications, Como, LNCS 488*. Springer-Verlag, April 1991.

14. Emil L. Post. Recursive unsolvability of a problem of Thue. *Journal of Symbolic Logic*, 13:1–11, 1947.

15. Alan M. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proc. London Math. Soc.*, 42(2):230–265, 1936.

16. Alan M. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proc. London Math. Soc.*, 43:544–546, 1937.