NEW DECISION ALGORITHMS FOR FINITELY PRESENTED COMMUTATIVE SEMIGROUPS

by

A. M. Ballantyne and D. S. Lankford

May 1979

ATP-59

This work was supported in part by the National Science Foundation Grant MSC77-20701.

NEW DECISION ALGORITHMS FOR FINITELY PRESENTED COMMUTATIVE SEMIGROUPS

A. M. Ballantyne and D. S. Lankford

MTP-4 May 1979

Department of Mathematics and Statistics
Louisiana Tech University
Ruston, Louisiana 71272

NEW DECISION ALGORITHMS FOR FINITELY PRESENTED COMMUTATIVE SEMIGROUPS

A. M. Ballantyne
University of Texas
Mathematics Department
Austin, Texas 78712

D. S. Lankford
Louisiana Tech University
Mathematics Department
Ruston, Louisiana 71272

May 1979
revised January 1980

ABSTRACT

A new solution of the uniform word problem for finitely presented commutative semigroups is constructed from a completion procedure for commutative-associative term rewriting systems. The completion procedure transforms a finite presentation into a uniformly terminating equivalence class term rewriting system which is Church-Rosser (terminates uniquely) and therefore decides equivalence of words in the given finitely presented commutative semigroup. Words are expressed in multiplicative exponential form, i.e., as finite vectors, so that fixed uniformly terminating Church-Rosser equivalence class term rewriting systems decide equivalence of words in constant space. Since the uniform word problem for finitely presented commutative semigroups requires exponential space on infinitely many instances, a Church-Rosser term rewriting system must be exponentially larger than its presentation for infinitely many presentations. This solution of the uniform word problem for finitely presented commutative semigroups, in addition to being conceptually simpler than previous solutions, is another small step towards the systematic application of uniformly terminating Church-Rosser term rewriting systems to the solved and open decision problems of algebra.

INTRODUCTION

This paper consists of five parts--introduction, background, main results, computer generated examples, and conclusions. In the introduction we discuss the origin of the methods of this paper and explain what we show. The section on background summarizes the facts about commutative-associative term rewriting systems which are needed to motivate the main results. Several computer generated examples are given which also illustrate the completion procedure.

The methods of this paper are applications of a general approach to constructing word problem decision algorithms for arbitrary abstract algebras. The general approach, known as completion, was developed by Knuth and Bendix [1970] for ordinary term rewriting systems and extended to equivalence class term rewriting systems by Huet [1977], Lankford and Ballantyne [1977a,b,c] and Stickel and Peterson [1977]. The general approach owes much to the pioneering work of Evans [1951] who was one of the first to demonstrate the remarkable effectiveness of applying the diamond lemma (Newman [1942]) to word problems. The subsequent developments mentioned above are primarily concerned with automatic and semi-automatic computer generation of word problem decision algorithms by methods based on the diamond lemma. Equivalence class rewriting methods have also been developed and used by Evans [1963a] and Treash [1969].

In this paper we restrict our attention to the class of term rewriting systems whose rewrite rules are formed from terms over one binary commutative-associative function and a finite number of constants.

A finitely presented commutative semigroup is defined by a finite number of <u>generators</u> (constants) and <u>relations</u> (axioms whose terms are constructed from generators and one binary commutative-associative function). For brevity we express relations in exponential multiplicative form with the commutative-associative operator omitted. For example, $(a, b, c, d; ac = abc^2, a^2b = c, bc = c^2)$ is a <u>presentation</u>. A total order, the <u>lexicographic order on vectors</u>, is defined on terms by $c_1^{m_1} \ldots c_k^{m_k} > c_1^{n_1} \ldots c_k^{n_k}$ iff $m_1 + \ldots + m_k > n_1 + \ldots + n_k$ or $m_1 + \ldots + m_k = n_1 + \ldots + n_k$, $m_1 = n_1$, $\ldots$, $m_{j-1} = n_{j-1}$, and $m_j > n_j$. The purpose of the total order is to insure a priori that term rewriting systems with left sides of rewrite rules larger than corresponding right sides are necessarily uniformly terminating, and that the completion procedure does not halt with incomparable relations. This particular total order is used because it facilitates our demonstration that the completion procedure halts uniformly on finitely presented commutative semigroups.

The completion procedure begins by expressing all relations of a given presentation as rewrite rules $L \longrightarrow R$ which satisfy $L > R$. (Trivial relations of the form $t = t$ are always deleted.) The completion procedure is based on an algorithm which decides the Church-Rosser (unique termination) property for uniformly terminating commutative-associative term rewriting systems. For the moment it is not necessary to know the precise details of the Church-Rosser decision algorithm, but only that it halts if the term rewriting system is Church-Rosser, and outputs a finite number of equations (relations) if not. Any new equations are expressed as rewrite rules $L \longrightarrow R$ satisfying $L > R$ and the process is iterated. For example, given the presentation above, the completion procedure proceeds as follows.

1.	$abc^2 \longrightarrow ac$	relations expressed as rewrite rules
2.	$a^2b \longrightarrow c$	
3.	$bc \longrightarrow c^2$	
4.	$ac^3 \longrightarrow ac$	rule 1 reduced by rule 3, rule 1 deleted
5.	$c^4 \longrightarrow c^2$	new equation expressed as rewrite rule
6.	$a^2c \longrightarrow c^3$	new equation expressed as rewrite rule

The completion procedure halts with the Church-Rosser commutative-associative term rewriting system consisting of rules 2 - 6. Equivalence of two words is decided by determining if their fully reduced forms are in the same commutative-associative equivalence class. For example, $a^2b^3c^2$ and bc^2 are equivalent words in the above commutative semigroup because both fully reduce to c^3.

The uniform word problem for finitely presented commutative semigroups is shown to be solvable by the completion procedure by showing that the completion procedure halts for any input presentation. The main results of this paper are two lemmas which reduce the uniform halting of the completion procedure to the fact that there is no infinite set of mutually incomparable vectors in N^m under the ordering $(x_1,\ldots,x_m) \geq (y_1,\ldots,y_m)$ iff $x_i \geq y_i$ for $i = 1,\ldots,m$.

BACKGROUND

In this section we summarize the facts about commutative-associative (hereafter abbreviated C-A) term rewriting systems which are needed to motivate the main results. We emphasize again that the methods of this paper are applications of much more general techniques. Many of the definitions and theorems of this paper appear in greater generality elsewhere, e.g., Huet [1977], Lankford and Ballantyne [1977a,b,c] and Stickel and Peterson [1977].

Terms are constructed from one binary C-A function f and a finite number of constants. A congruence relation, denoted $\equiv$, is defined on terms by $t \equiv u$ iff $t = u$ is an equality consequence of the two axioms $f(x,y) = f(y,x)$ and $f(f(x,y),z) = f(x,f(y,z))$. Let $[t]$ denote the equivalence class of t under $\equiv$. A C-A term rewriting system, denoted $\mathcal{R}$, is a finite set of expressions $[L] \longrightarrow [R]$ where L and R are terms. We say that $[u]$ is an immediate reduction of $[t]$ (relative to $\mathcal{R}$), denoted $[t] \longrightarrow [u]$, in case there is a member $[L] \longrightarrow [R]$ of $\mathcal{R}$ and members t', u', L', and R' respectively such that u' is the result of replacing one occurrence of L' in t' by R'. A C-A term rewriting system is uniformly terminating in case there is no infinite sequence $[t_1] \longrightarrow [t_2] \longrightarrow [t_3] \longrightarrow \ldots$ of immediate reductions. We say that $[t]$ is irreducible in case $[t]$

has no immediate reductions. A uniformly terminating C-A term rewriting system is Church-Rosser (uniquely terminating) in case for any two sequences $[t] \longrightarrow \ldots \longrightarrow [u]$ and $[t] \longrightarrow \ldots \longrightarrow [v]$ with $[u]$ and $[v]$ irreducible, $[u] = [v]$.

<u>Diamond Lemma</u> (Newman [1942]) A uniformly terminating relation $\longrightarrow$ is Church-Rosser iff for each t, u, and v, if $t \longrightarrow u$ and $t \longrightarrow v$ then there is an irreducible w such that $u \longrightarrow \ldots \longrightarrow w$ and $v \longrightarrow \ldots \longrightarrow w$.

<u>Proof</u> See Huet [1977].

We haven't said what an arbitrary uniformly terminating or Church-Rosser relation actually is, but those notions should be clear after a little reflection on the definitions above. As we have said, the completion procedure upon which the solution of the uniform word problem for finitely presented commutative semigroups is built is in turn built on a decision algorithm for the Church-Rosser property.

The Diamond lemma is the basis of Church-Rosser decision algorithms (when they exist), see, e.g., Huet [1977], Knuth and Bendix [1970], Lankford and Ballantyne [1977a,c], and Stickel and Peterson [1977].

Before we get involved in the details of how a Church-Rosser decision algorithm for finitely presented commutative semigroups is constructed from the Diamond Lemma, we explain why Church-Rosser decision algorithms are interesting to those who study word problem decision algorithms for arbitrary abstract algebras. The crux of the matter is that a uniformly terminating Church-Rosser equivalence class term rewriting system $\mathcal{R}$ with a decidable equivalence relation and for which irreducibility is decidable decides the word problem for $\mathcal{E}(\mathcal{R})$, the equational theory whose axioms are obtained from $\mathcal{R}$ by replacing "$\longrightarrow$" by "=" together with the equations which define the equivalence relation. For uniformly terminating Church-Rosser $\mathcal{R}$, two words are equal in $\mathcal{E}(\mathcal{R})$ iff [t] and [u] fully reduce to equal equivalence classes. If there were no interesting examples of uniformly terminating Church-Rosser term rewriting systems, this characterization of decidable word problems would be only a mildly interesting mathematical result. But many of the common equational theories with decidable word problems are decidable by uniformly terminating Church-Rosser term rewriting systems, e.g., the free group on no generators and no relations, central groupoids, L-R systems, see Knuth and Bendix [1970]; the free Abelian group and the free commutative ring on no generators and no relations, see Lankford and Ballantyne [1977c] and Stickel and Peterson [1977]; a fragment of recursive function theory (which is a new subclass of recursive functions with a decidable word problem), see Degano and Sirovich [1979]; finitely presented loops (quasigroups, groupoids, inverse property loops, etc.), see Evans [1951]; finitely presented trees, see Evans [1963a]; finitely presented Steiner loops, see Treash [1969]; finitely presented groups whose word problems are solvable by Dehn's algorithm,[1] see Bucken [1979]; and finitely presented commutative semigroups (this paper).

Throughout the remainder of this paper we think of words in a finitely presented commutative semigroup interchangeably as terms over one binary C-A function and a finite number of constants $c_1, \ldots, c_k$, denoted in multiplicative form $c_1^{m_1} \ldots c_k^{m_k}$, or as k-tuples (vectors) of non-negative integers $(m_1, \ldots, m_k)$. The term formulation of words is necessary to apply the term rewriting methods, while the vector formulation seems to be the most efficient for computer implementations.

In the vector formulation, two C-A equivalence classes are equal iff the representative vectors are equal. Irreducibility also has a simple characterization--a word $(p_1, \ldots, p_k)$ is reducible by a rewrite rule $(m_1, \ldots, m_k) \longrightarrow (n_1, \ldots, n_k)$ iff $p_i \geq m_i$ for $i = 1, \ldots, k$. Moreover, immediate reductions (when they exist) are easy to compute, e.g., $(p_1 - m_1 + n_1, \ldots, p_k - m_k + n_k)$. Thus, it is clear that there is an algorithm, which we denote $*$, that reduces each equivalence class $[t]$ to an irreducible $[t]^*$.

1. this is not quite accurate, and will be revised

Given two C-A rewrite rules $(m_1,\ldots,m_k) \longrightarrow (n_1,\ldots,n_k)$ and $(p_1,\ldots,p_k) \longrightarrow (q_1,\ldots,q_k)$, they produce a <u>critical pair</u> $(a_1,\ldots,a_k)$, $(b_1,\ldots,b_k)$ provided the vector $(x_1,\ldots,x_k) = (\min(m_1,p_1),\ldots,\min(m_k,p_k))$ is not the zero vector, where

$$(a_1,\ldots,a_k) = (n_1+p_1-x_1,\ldots,n_k+p_k-x_k) \quad \text{and}$$
$$(b_1,\ldots,b_k) = (q_1+m_1-x_1,\ldots,q_k+m_k-x_k) \quad .$$

Thus, critical pairs are formed from rewrite rules by substituting on left sides with "maximum overlap." For example, the critical pair of $abc^2 \longrightarrow ac$ and $a^2b \longrightarrow c$ is a^2c, c^3. Each pair of rewrite rules has at most one critical pair. The following Church-Rosser theorem shows that the critical pairs are the vector representations of terms u and v from the diamond lemma which must reduce to a single term w; and, moreover, that not every pair u, v must be tested to verify that the Church-Rosser property holds, but just the critical pairs.

<u>Church-Rosser Theorem</u> A uniformly terminating C-A term rewriting system (over one binary C-A function and a finite number of constants) is Church-Rosser iff for each critical pair X, Y, $X^* = Y^*$.

Before we prove the Church-Rosser Theorem we point out that it at once provides us with a Church-Rosser decision algorithm--form all critical pairs and see if they reduce to the same vector. For simple examples, like the one given in the introduction, the algorithm can be performed by hand. The example critical pair above satisfies the test since $(a^2c)^* = c^3 = (c^3)^*$. This is not actually one of the critical pairs that must be tested for the example in the introduction, since it is rules 2 - 6 which are Church-Rosser. However, there are only 14 critical pairs that must be tested to show that the example is Church-Rosser, and the reader may wish to check a few to get a better idea of what the Church-Rosser test involves.

<u>Proof</u> ($\Longrightarrow$) It is easy to show that there exists a vector Z such that $Z \longrightarrow X$ and $Z \longrightarrow Y$, hence $Z \longrightarrow \ldots \longrightarrow X^*$ and $Z \longrightarrow \ldots \longrightarrow Y^*$. Since the term rewriting system is Church-Rosser, it follows that $X^* = Y^*$.

($\Longleftarrow$) Let $T \longrightarrow U$ and $T \longrightarrow V$. If the rewrite rules that produce these two immediate reductions do not "interact", then there is clearly a W such that $U \longrightarrow W$ and $V \longrightarrow W$, hence $U \longrightarrow \ldots \longrightarrow W^*$ and $V \longrightarrow \ldots \longrightarrow W^*$. If they do "interact", let M_1 be the common part, let $L_1 \longrightarrow R_1$ and $L_2 \longrightarrow R_2$ be the two rewrite rules, let $L_1 = L_1'M_1M_2$ and $L_2 = L_2'M_1M_2$ where M_1M_2 is the maximum overlap, and observe that $T = AL_1'L_2'M_1M_2M_2$ so that $U = AM_2L_2'R_1$ and $V = AM_2L_1'R_2$. Since the critical pairs test holds, it follows that $(L_2'R_1)^* = (L_1'R_2)^*$, hence $U \longrightarrow \ldots \longrightarrow$

$(AM_2(L_2{}'R_1)^*)^*$ and $V \longrightarrow \ldots \longrightarrow (AM_2(L_2{}'R_1)^*)^*$. This completes the proof of the Church-Rosser Theorem.

This Church-Rosser theorem is a new result, but we include it in the section on background because it is suggested by the more general results of Lankford and Ballantyne [1977] and Stickel and Peterson [1977].

We conclude the section on background with a description of the completion algorithm.

<u>Completion Algorithm</u>

1. Express a given presentation as a uniformly terminating set of rewrite rules $\mathcal{R}$. (Relations are converted to rewrite rules via the lexicographic order on vectors, see the introduction.)
2. Reduce $\mathcal{R}$.
3. Generate and test critical pairs of $\mathcal{R}$.
   - (i) If $\mathcal{R}$ is Church-Rosser, halt.
   - (ii) If not, add new rewrite rules (the non-equal critical pairs ordered by the lexicographic order on vectors) to $\mathcal{R}$ and go to step 2.

As one can see, the informal description of the completion procedure given in the introduction is accurate only to a first approximation. The aspect not previously mentioned is step 2, where $\mathcal{R}$ is reduced. This aspect is crucial to our proof that the completion procedure halts uniformly.

$\mathcal{R}$ is reduced in the completion algorithm above as follows. Each rewrite rule $L \longrightarrow R$ is considered in turn. If L or R is not irreducible relative to $\mathcal{R} - \{L \longrightarrow R\}$, form L^{**} and R^{**} (where $**$ is reduction to irreducible form by $\mathcal{R} - \{L \longrightarrow R\}$), and return $L^{**} \longrightarrow R^{**}$ or $R^{**} \longrightarrow L^{**}$ depending on whether $L^{**} > R^{**}$ or $R^{**} > L^{**}$ respectively (if $L^{**} = R^{**}$, reduction is continued on $\mathcal{R} - \{L \longrightarrow R\}$). Eventually $\mathcal{R}$ is "reduced" so that no rewrite rule is reducable by any of the others. (When a reduced rewrite rule is returned to $\mathcal{R}$, the rule it is obtained from is deleted.) An example of the completion algorithm is given in the introduction and in the section on computer generated examples.

MAIN RESULTS

Lemma 1 (Hack [1974]) Any set of mutually incomparable vectors in N^m is finite.

Lemma 2 If the completion procedure did not terminate uniformly on finitely presented commutative semigroups, then there would be an infinite set of mutually incomparable vectors in N^m.

Proof Suppose the completion procedure did not terminate for some presentation, and let $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3, \ldots$ be the infinite sequence of rewrite rules produced at step 2 of the completion procedure. The rewrite rules are totally ordered by $L_1 \longrightarrow R_1 > L_2 \longrightarrow R_2$ iff $L_1 > L_2$ or $L_1 = L_2$ and $R_1 > R_2$. Eventually in some $\mathcal{R}_i$ the least rewrite rule is produced. Eventually in some $\mathcal{R}_j$, $i \leq j$, the next least rewrite rule is produced. Continuing in this way, we get the infinite sequence of rewrite rules $L_1 \longrightarrow R_1$, $L_2 \longrightarrow R_2$, ... and hence an infinite sequence $L_1, L_2, \ldots$ whose terms are mutually incomparable, which is impossible.

Corollary The completion algorithm solves the uniform word problem for finitely presented commutative semigroups.

COMPUTER GENERATED EXAMPLES

The completion algorithm was programmed in LISP on a DEC 10 at the University of Texas at Austin. The program was about three pages of LISP code. Several random presentations were given to the program, including the example in the introduction. Since the example in the introduction illustrates most aspects of the completion procedure, we show its derivation in detail.

Example 1 Presentation: $ac = abc^2$, $a^2b = c$, $bc = c^2$

1. $abc^2 \longrightarrow ac$
2. $a^2b \longrightarrow c$
3. $bc \longrightarrow c^2$

In steps 1 - 3 above the presentation has been expressed as a uniformly terminating set of rewrite rules, step 1 of the completion algorithm. Now step 2 is performed, and rule 1 is reduced by rule 3, so rule 1 is deleted and a new rule is added.

4. $ac^3 \longrightarrow ac$

Now step 3 of the completion algorithm is performed, three critical pairs are formed, and two fail the critical pairs test and are added as rewrite rules.

5. $a^2c^2 \longrightarrow c^2$ by 2 and 3
6. $c^4 \longrightarrow c^2$ by 2 and 4

The procedure returns to step 2, but 2 - 6 are already reduced, so the procedure returns to step 3 where six critical pairs are formed, and one fails the critical pairs test and is added as a rewrite rule.

7. $a^2c \longrightarrow c^3$ by 3 and 5 (also by 4 and 5)

The procedure returns to step 2, where rule 5 is deleted. Rules 2, 3, 4, 6 and 7 are tested again in step 3 of the completion procedure and found to be Church-Rosser. The actual computer program does not not follow these precise steps. The difference is that instead of generating all critical pairs at step 3 of the stated completion procedure, when a critical pair is generated which fails the test it is immediately added to $\mathcal{R}$ and the procedure is restarted at step 2. Empirical evidence suggests that this is often a more efficient procedure than the one stated. In this example, rule 5 was not generated by the computer program.

<u>Example 2</u> Presentation: $a^2b^4c^5d^3 = ab^3c^2d^2$, $a^8bc^3d^4 = ab^8c^4d^2$, $a^2b^2c^2d^2 = c^2d$, $ab^8c^3 = b^2d^3$, $a^7d^7 = c^8$.

Church-Rosser term rewriting system:

1. $a^2b^2c^2d^2 \longrightarrow c^2d$
2. $ab^8c^3 \longrightarrow b^2d^3$
3. $a^7d^7 \longrightarrow c^8$
4. $a^6bc^3d^3 \longrightarrow b^4cd^5$
5. $ab^5c^2d^5 \longrightarrow a^5c^4d^2$
6. $a^5c^6d^2 \longrightarrow a^6bc^3d^2$
7. $a^2b^5cd^5 \longrightarrow a^6c^3d^2$
8. $a^7c^3d^2 \longrightarrow b^2c^4d^4$
9. $a^{10}c^7d \longrightarrow a^2bc^5d^2$

NOTE: This example is wrong because of a "bug" in Mike's program, see Addendum 2/13/85.

10. $a^2bc^6d^3 \longrightarrow ac^3d^2$

11. $a^7c^8d \longrightarrow a^9b^2c^2d$

12. $b^6c^4d^5 \longrightarrow a^3c^3d$

13. $a^4bc^9d \longrightarrow a^6b^3c^3d$

14. $a^5b^2c^6d \longrightarrow a^6b^3c^3d$

15. $abc^{15}d^2 \longrightarrow a^2b^2c^2d$

16. $ab^7c^3d^4 \longrightarrow a^4bc^8d$

17. $abc^9d^3 \longrightarrow abc^3d^2$

18. $a^{10}bc^2d \longrightarrow abc^3d^2$

19. $a^4bc^8d^2 \longrightarrow a^4bc^2d$

20. $a^7bc^3d \longrightarrow b^3c^4d^3$

21. $b^7c^3d^5 \longrightarrow a^3bc^8d^2$

22. $a^2b^4cd^6 \longrightarrow b^2cd^5$

23. $abc^4d^5 \longrightarrow a^2b^2cd^5$

24. $b^5cd^7 \longrightarrow a^4c^3d^4$

25. $a^7bc^2d^3 \longrightarrow b^3c^3d^5$

26. $b^2c^5d^2 \longrightarrow ab^3c^2d^2$

CONCLUSIONS

There are at least seven other solutions of the uniform word problem for finitely presented commutative semigroups--see Biryukov [1967], Emelichev [1958], Hermann [1926], Malcev [1958], Rabin [1965], Simmons [1980] and Tseiten (Redei [1965]). A modern treatment of Hermann's work, with some corrections, is found in Seidenberg [1974]. It is said that Emelichev [1958] and Malcev [1958] were the first to explicitly state the decidability of this problem, though in retrospect the solution can be seen to be contained in the work of König [1903] and Hermann [1926]. The connection with Hermann's work on polynomial ideals is made explicit by Cardoza, et al. [1976], who also established the exponential complexity of the uniform word problem for finitely presented commutative semigroups. From comments in Cardoza [1975] we suspect there is a close relationship between our approach and Biryukov's, but we also suspect there are significant differences since Biryukov's approach seems to be a classical basis construction, while ours might be thought of as a construction of the "fundamental identities". In any case, we think it is fair to say that of the commonly known solutions, ours is conceptually simpler than the others.

As we have said, one of the main reasons for studying uniformly terminating Church-Rosser term rewriting systems is that the completion procedures offer a systematic approach to decision problems of algebra. The list of problems solved in this way (see the background section) is almost certain to grow (finitely presented Abelian groups?, nilpotent groups?, commutative rings? , boolean algebras?, etc.). And for each new application, we anticipate that the classical solution will be modestly extended (Bücken [1979] properly extended Dehn's algorithm, and we could extend the results of this paper to finitely presented commutative semigroups "with operators"). We also think that many of the other commonly studied decision problems for abstract algebras can be profitably analyzed by extensions of the term rewriting methods (e.g., the isomorphism problem, the triviality problem, the finiteness problem, the subalgebra membership problem, boundedness problems, regularity problems, etc.). Indeed, a version of the completion algorithm has been used to solve the isomorphism problem for finitely presented loops, quasigroups, groupoids and loops, see Evans [1963b, 1980].

ACKNOWLEDGMENT

We would like to express our appreciation to E. Cardoza, M. Davis, T. Evans and J. Schwartz for their helpful assistance.

BIBLIOGRAPHY

Biryukov, A. P. [1967] Some algorithmic problems for finitely defined commutative semigroups. Siberian Mathematics Journal, Vol. 8, 384-391.

Bücken, H. [1979] Reduction systems and small cancellation theory. Proc. Fourth Workshop on Automated Deduction, Austin, Texas, Feb. 1-3, 53-59.

Cardoza, E. W. [1975] Computational complexity of the word problem for commutative semigroups. MAC Technical Memo. 67, MIT, Cambridge, Mass. 02139, master's thesis.

Cardoza, E., Lipton, R. and Meyer, A.R. [1976] Exponential space complete problems for Petri nets and commutative semigroups. Proc. 8th Annual ACM Symposium on Theory of Computing, 50-54.

Degano, P. and Sirovich, F. [1979] On deciding equivalence for a class of primitive recursive functions. Consiglio Nazionale delle Ricerche, Instituto di Elaborazione delle Informazione, Via S. Maria, 46-56100 PISA, Italy.

Emelichev, V. A. [1958] Commutative semigroups with one defining relation. Shuya Gosudarstvennyi Pedagogicheskii Institut Uchenye Zapiski, Vol. 6, 227-242.

Evans, T. [1951] On multiplicative systems defined by generators and relations. I, Normal form theorems. Proc. Cambridge Philos. Soc. 47, 637-649.

Evans, T. [1963a] A decision problem for transformations of trees. Canadian J. Math. 15, 584-590.

Evans, T. [1963b] The isomorphism problem for some classes of multiplicative systems. Trans. Amer. Math. Soc. 109, 303-312.

Evans, T. [1980] Some solvable word problems. Word Problems II, eds., W. Boone and G. Higman, North-Holland, Amsterdam, 87-100.

Hack, M. [1974] Decision problems for Petri nets and vector addition systems. Computational Structures Group Memo 95, Project MAC, MIT, Cambridge, Mass. 02139.

Hermann, G. [1926] Die frage der endlich vielen schritte in der theorie der polynomideale. Math. Ann., Vol. 95, 736-788.

Huet, G. [1977] Confluent reductions: abstract properties and applications to term rewriting systems. In Proc. 18th Symposium on Foundations of Computer Science, Providence, R.I., and also IRIA Laboria, Rocquencourt, B.P. No. 105, 78150 Le Chesnay, France, Rapport de Recherche No. 283.

Knuth, D. E. and Bendix, P. B. [1970] Simple word problems in universal algebras. Proc. Computational Problems in Abstract Algebras, J. Leech, ed., Pergamon Press, 263-297.

König, J. [1903] Einleitung in die Allgemeine Theorie der Algebraischen Groszen. B. B. Teubner, Leipzig

Lankford, D. S. and Ballantyne, A. M. [1977a] Decision procedures for simple equational theories with commutative axioms: complete sets of commutative reductions. Univ. of Texas, Math. Dept., Automatic Theorem Proving Project, Austin, Texas 78712, report ATP-35.

Lankford, D. S. and Ballantyne, A. M. [1977b] Decision procedures for simple equational theories with permutative axioms: complete sets of permutative reductions. Univ. of Texas, Math. Dept., Automatic Theorem Proving Project, Austin, Texas 78712, report ATP-37.

Lankford, D. S. and Ballantyne, A. M. [1977c] Decision procedures for simple equational theories with commutative-associative axioms: complete sets of commutative-associative reductions. Univ. of Texas, Math. Dept., Automatic Theorem Proving Project, Austin, Texas 78712, report ATP-39.

Malcev, A. I. [1958] On homomorphisms of finite groups. Ivano Gosudarstvennyi Pedagogicheskii Institut Uchenye Zapiski, Vol. 18, 49-60.

Newman, M. H. A. [1942] On theories with a combinatorial definition of equivalence. Annals of Math., Vol. 43, 223-243.

Rabin, M. [1965] Logic, Automata, and Computability Conference, Oriskany, New York.

Redei, L. [1965] The Theory of Finitely Generated Commutative Semigroups. Pergamon Press, New York.

Seidenberg, A. [1974] Constructions in algebra. Trans. Amer. Math. Soc., Vol. 197, 272-313.

Simmons, H. [1980] The word and torsion problems for commutative Thue systems. Word Problems II, eds. W. Boone and G. Higman, North-Holland Pub. Co., 1980, 395-400.

Stickel, M. and Peterson, G. [1977] Complete sets of reductions for equational theories with complete unification algorithms. RCA David Sarnoff Research Center, Princeton, New Jersey 08540, to be revised.

Treash, C. [1969] Inverse property loops and related Steiner triple systems. Emory U., Ph.D. thesis, 1969.

ADDENDUM

March 1980

Finitely presented commutative semigroups do not in general possess complete T-unification algorithms. For example, the commutative semigroup defined by $ab \longrightarrow b$ does not have a complete T-unification algorithm since the terms ax and a have infinitely many "mgu's": $\{b/x\}$, $\{bb/x\}$, $\{bbb/x\}$, ...

We have subsequently learned that a similar overlap method has been used by G. M. Bergman, "The diamond lemma for ring theory," Advances in Math. 29 (1978), 178-218 to solve the uniform word problem for finitely presented commutative semigroups. However, Bergman does not implement his method on a computer, and his proof of uniform termination of the completion procedure is quite different. It appears that our method of establishing uniform termination of the completion procedure may have some advantages over Bergman's method since, for example, it is not difficult to extend the results of our paper to the case of finitely presented algebras satisfying $f(x,f(y,z)) = f(y,f(x,z))$ as well as finitely presented semigroups satisfying the above identity.

POSTSCRIPT

7/23/1980

We have recently learned that Hack's Lemma might just as well be called Dickson's or Hilbert's Lemma. This we learned from a comment in Biryukov [1967] which led to a comment in Redei [1965] which led to L. Dickson's paper "Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors," Amer. J. Math. XXXV (1913), 413-422. There Dickson's Lemma A or Lemma B yields Hack's Lemma as an easy Corollary. Moreover, Dickson points out that his Lemma A is an easy Corollary of Hilbert's Basis Theorem. Thus the differences between the termination proofs of this article and Bergman [1978] are superficial, though these superficially different approaches may each have their advantages in different settings.

ADDENDUM

2/13/85

Example 2 is incorrect because of a bug in Mike Ballantyne's program. This was suspected in the summer of 1983 when programs written for the Abelian group uniform word problem at Louisiana Tech generated different complete sets than were gotten by a modification of the commutative semigroup program which Mike had done. Mike subsequently determined in the fall of 1983 that his commutative semigroup program was indeed incorrect, but Mike never communicated the correction to me. The error was found independently by D. Kapur in the fall of 1984. The following correct complete set for example 2 was generated by the Gröbner basis program of Kandri-Rody and Kapur:

1. $ac^2d \longrightarrow c^2d$	4. $c^2d^2 \longrightarrow c^2d$	7. $c^{10} \longrightarrow c^2d$
2. $bc^2d \longrightarrow c^2d$	5. $b^2d^4 \longrightarrow c^2d$	8. $ab^8c^3 \longrightarrow b^2d^3$
3. $c^3d \longrightarrow c^2d$	6. $b^2c^8 \longrightarrow c^2d$	9. $a^7d^7 \longrightarrow c^8$