Some Properties of Finite Special String-rewriting Systems

LOUXIN ZHANG

Department of Computer Science, University of Waterloo
Waterloo, Canada, N2L 3G1

(Email: lzhang@neumann.waterloo.edu)

(Received 24 April 1992)

This paper investigates decision problems of finite, special string-rewriting systems. There are two main results. The first one is that the word problem for a finite, special string-rewriting system T on alphabet A is reducible to its restricted version: given a word w, is w congruent to any fixed element z on A? Another is a Markov type theorem: a property P is undecidable for finite, special string-rewriting systems if P implies any fixed Markov property of finitely presented special monoids and there exists a finite, special string-rewriting system R on alphabet C with the property that a finite, special string-rewriting system T on A has P whenever M(A; T) is isomorphic to M(C; R).

1. Introduction

String-rewriting systems, also known as semi-Thue systems, have extensively been studied in computability theory, combinatorial (semi-)groups, and formal language theory. They are used to present semigroups and groups ( Lallement, 1979), and to specify formal languages as unions of congruence classes (Nivat & Benois, 1971). Furthermore, each string-rewriting system on alphabet A can be interpreted as a term-rewriting system by simply regarding each symbol a \in A as an unary function symbol. Thus results on string-rewriting systems can give some valuable insight on what may be done by applying term-rewriting systems in diverse areas such as abstract data types, automated theorem proving, and computer algebra.

In this paper, we focus our attention on the computational aspects of finite, special string-rewriting systems. Here a string-rewriting system T is special if the right-hand side of each rule of T is the empty word. A monoid is special if it can be presented by some special string-rewriting system. Groups are special monoids. But, there exist special monoids that are not groups, one of which is the bicyclic monoid presented by T = { (ab, e) } over generators a and b. Adjan(1966) presented a detailed study of special monoids.

The paper is structured as follows. In Section 2, we restate the basic notions and
notation on string-rewriting systems, Thue congruences, and monoid-presentations that we shall use throughout the paper.

In Section 3, we present some basic results. Given a finite, special string-rewriting system \( T \), we define a well-behaved string-rewriting system \( R \) using the set of minimal words, which consists of all invertible words modulo \( T \) with lengths \( \leq \max_{(i,s) \in \mathcal{P}} |i| \), and show that \( R \) is confluent and equivalent to \( T \) (Proposition 3.4). This technical result is used in Section 4.

The word problem for a string-rewriting system \( T \) on alphabet \( A \) can be stated as follows: given two words \( u, v \in A^* \), decide in a finite number of steps whether \( u \) and \( v \) are congruent modulo \( T \). If \( M(A; T) \) happens to be a group, then the word problem for \( T \) is reducible to the restricted version of the word problem (commonly known in the literature as the special word problem): given a word \( w \in A^* \), is \( w \) congruent to the empty word \( e \)? In Section 4, the above result is extended to finite, special string-rewriting systems (Theorem 4.7).

In Section 5, we prove a Markov type theorem (Theorem 5.3): a property is undecidable for finite, special string-rewriting systems if it satisfies the following two conditions: (1) it implies a given fixed Markov property of finitely presented special monoids; (2) there exists a finite, special string-rewriting system \( R \) on an alphabet \( B \) such that a finite, special string-rewriting system \( T \) on \( A \) has the property whenever \( M(A; T) \) is isomorphic to \( M(B; R) \). This result generalises a result of O'Dunlaing (1983). Applying the above result, we obtain that the following problems are undecidable in general: (1) is a finitely presented special monoid a group (Narendran et al)? (2) does the Knuth-Bendix completion algorithm terminate on a finite, special string-rewriting system (Proposition 5.5 (1))? (3) is a finite, special string-rewriting system equivalent to a finite, Noetherian string-rewriting system that is confluent on a given congruence class (Proposition 5.5 (2))?

2. Preliminaries

Here we provide formal definitions of string-rewriting systems and related notions. For additional information and comments regarding the various notions introduced, the reader is asked to consult the excellent survey paper Book (1987). For a general discussion of decidability and algorithms, see the book by Davis (1958).

Let \( A \) be a finite alphabet, and let \( A^* \) be the free monoid generated by \( A \). We write \( A^+ = A^* - \{e\} \), where \( e \) is the identity of \( A^* \). For an element \( w \in A^* \), the length of \( w \), denoted by \( |w| \), is defined as follows: \( |e| = 0 \), and \( |wa| = |w| + 1 \) for \( w \in A^* \) and \( a \in A \).

For a set \( S \subseteq A^* \) and \( u \in A^* \), the left and right quotients of \( S \) with respect to \( u \) are defined respectively as \( u^{-1}S = \{v \in A^*|uv \in S\} \) and \( Su^{-1} = \{v \in A^*|vu \in S\} \). For \( S_1, S_2 \subseteq A^* \), the product of \( S_1 \) and \( S_2 \) is defined by \( S_1S_2 = \{uv \in A^*|u \in S_1, v \in S_2\} \).

A string-rewriting system \( T \) on an alphabet \( A \) is a set of ordered pairs of elements of \( A^* \), i.e., \( T \subseteq A^* \times A^* \), the elements of which are called (rewrite) rules. For a string-rewriting system \( T \), \( \text{dom}(T) = \{l|\exists r \in A^*: (l, r) \in T\} \), and \( \text{range}(T) = \{r|\exists l \in A^*: (l, r) \in T\} \).
The system $T$ is called length-reducing, if $|l| > |r|$ holds for each rule $(l, r) \in T$; it is called special, if it is length-reducing and $\text{range}(T) = \{e\}$.

A string-rewriting system $T$ on $A$ induces a number of binary relations on $A^*$, the most fundamental one of which is the single-step reduction relation $\rightarrow_T$: for $u, v \in A^*$, $u \rightarrow_T v$ if and only if $\exists x, y \in A^*, \exists (l, r) \in T : u = xly$ and $v = xry$. Its reflexive and transitive closure $\rightarrow^*_T$ is the reduction relation induced by $T$, and its reflexive, symmetric, and transitive closure $\rightarrow^{**}_T$ is a congruence on $A^*$, called the Thue congruence generated by $T$. For $w \in A^*$, $[w]_T$ denotes the congruence class $\{v \in A^* | v \rightarrow^{**}_T w\}$. The set $\{[w]_T | w \in A^*\}$ of congruence classes forms a monoid $M(A; T)$ under the operation $[u]_T \cdot [v]_T = [uv]_T$ with identity $[e]_T$, which is uniquely determined (up to isomorphism) by $A$ and $T$. Therefore, whenever a monoid $M$ is isomorphic to the monoid $M(A; T)$, we call the ordered pair $(A; T)$ a (monoid) presentation of $M$ with generators $A$ and defining relations $T$. If both $A$ and $T$ are finite, $M$ is finitely presented.

If $u, v \in A^*$ are such that $u \rightarrow_T v$, then we say that $u$ reduces to $v$, $u$ is an ancestor of $v$, and $v$ is a descendant of $u$ modulo $T$. If there is no word $v$ such that $u \rightarrow_T v$, then $u$ is irreducible; otherwise, it is reducible modulo $T$. Finally, for $U \in A^*$, let $< u >_T = \{w \in A^* | w \rightarrow^*_T u\}$.

A finite string-rewriting system $T$ on $A$ is called
— Noetherian if there exists no infinite sequence of reductions of the form $x_1 \rightarrow_T x_2 \rightarrow_T \cdots$;
— confluent if, for all $u, v, w \in A^*$, $u \rightarrow_T v$, and $u \rightarrow_T w$ imply that there exists some $z \in A^*$ such that $v \rightarrow_T^* z$ and $w \rightarrow_T^* z$.
— confluent on $[x]_T$ for some $x \in A^*$ if, for all $u, v, w \in [x]_T$, $u \rightarrow_T v$ and $u \rightarrow_T w$ imply that there exists some $z \in [x]_T$ such that $v \rightarrow_T^* z$ and $w \rightarrow_T^* z$.

If a finite, Noetherian string-rewriting system $T$ is confluent on the congruence class $[w]_T$, then $[w]_T$ contains a unique irreducible word $w_0$, which can then be taken as the normal form of this class, and $[w]_T = < w_0 >_T$; if it is confluent, every congruence class modulo $T$ has a normal form.

The property of being confluent is decidable for finite Noetherian string-rewriting systems. Actually, the following well-known result holds.

**Theorem 2.1.** (Nivat & Benois, 1971) Let $T$ be a finite string-rewriting system on $A$. Suppose $T$ is Noetherian. Then, $T$ is confluent if and only if, for all words $x, y, z, u, v \in A^*$,

1. if $(xy, u)$ and $(yz, v)$ are rules of $T$, then there is a $w \in A^*$ such that $xv \rightarrow_T^* w$ and $uz \rightarrow_T^* w$, and
2. if $(xyz, u)$ and $(y, v)$ are rules of $T$, then there is a $w \in A^*$ such that $u \rightarrow_T^* w$ and $xvz \rightarrow_T^* w$.

However, the property of being confluent on a given congruence class is much harder than the property of being confluent for finite Noetherian string-rewriting systems. This
property is undecidable even for finite, length-reducing string-rewriting systems (Otto, 1987).

3. Some Basic Results

Let $T$ be a finite string-rewriting system on an alphabet $A$. The word $z$ is left (right) invertible modulo $T$, if there exists a $y \in A^*$ such that $yz(xy) \rightarrow_T e$; it is invertible modulo $T$ if it is both left and right invertible. Hence $z$ is (left, right) invertible modulo $T$ if and only if $[z]_T$ is a (left, right) unit of the monoid $M(A; T)$ presented by $(A; T)$. Obviously, the product $u_1u_2$ of two invertible words $u_1$ and $u_2$ is invertible. Furthermore, we have the following result.

**Proposition 3.1.** Let $T$ be a finite string-rewriting system on $A$, and let $u, v, w \in A^*$. If $uv$ and $vw$ are invertible modulo $T$, then $u, v,$ and $w$ are also invertible modulo $T$.

**Proof.** Since both $uv$ and $vw$ are invertible modulo $T$, there exist two words $z$ and $y$ on $A$ such that $xuv \rightarrow_T yvw \rightarrow_T vwy \rightarrow_T e$, which implies that $v$ is invertible modulo $T$. Taking $z = yv$, we have $zw = yvw \rightarrow_T e$ and $wz = wyv \rightarrow_T (xuv)wyv \rightarrow_T e$. So $w$ is invertible modulo $T$. By symmetry, $u$ is invertible modulo $T$. □

Let $T$ be a finite, special string-rewriting system on $A$. We may assume that $dom(T) = \{l_1, l_2, \ldots, l_k\}$. A nonempty invertible word is called a minimal word if its length does not exceed $\max_{1 \leq i \leq k} |l_i|$, and none of its proper prefixes is invertible modulo $T$. It can easily be seen that the set of all minimal words forms a biprefix code $C$, i.e., $u^{-1}C = Cu^{-1} = \{e\}$ for all $u \in C$. Since each $l_i$ is an invertible word and $|l_i| \leq \max_{1 \leq j \leq k} |l_j|$, we have $l_i \in C^*$. Since $C$ is a biprefix code, $l_i$ can be uniquely decomposed into minimal factors:

$$l_i = l_{i,1}l_{i,2}\cdots l_{i,n_i} \quad (3.1)$$

for $i = 1, 2, \ldots, k$. For $p \leq n_i$, we set $\Delta_{i,p} = \{x \in C|x \rightarrow l_{i,p}\}$. Let $\Delta = \bigcup_{1 \leq i \leq k} \bigcup_{1 \leq p \leq n_i} \Delta_{i,p}$. As a subset of $C$, $\Delta$ is also a biprefix code.

**Proposition 3.2.** Let $T$ and $\Delta$ be defined as above, and let $x, y, z \in A^*$. If $xy \in \Delta^*$ and $yz \in \Delta^*$, then $x, y,$ and $z$ are also in $\Delta^*$.

**Proof.** $xy \in \Delta^*$ implies that $xy = u_1u_2\cdots u_k$, where $u_i \in \Delta$ for each $i, 1 \leq i \leq k$. Thus we have $x = u_1u_2\cdots u_{i-1}u'_i$ and $y = u''_iu_{i+1}\cdots u_k$, where $u'_i \in A^*$ and $u''_i \in A^+$ are such that $u_i = u'_iu''_i$. Since $u_i = u'_iu''_i$ and $u''_iu_{i+1}\cdots u_kx = yz$ are invertible modulo $T$, by Proposition 3.1, $u'_i$ is also invertible modulo $T$, which in turn implies that $u''_i = u_i$ since $u_i \in \Delta$. Therefore, $u'_i = e$, $x \in \Delta^*$ and $y \in \Delta^*$. Furthermore, since $\Delta$ is a biprefix code, $y \in \Delta^*$ and $yz \in \Delta^*$ imply that $z \in \Delta^*$. □

**Proposition 3.3.** Let $T$ and $\Delta$ be defined as above, let $x, y \in A^*$ and let $u, v \in \Delta^*$ are such that $u \leftrightarrow_T v$ and $|u| \geq |v|$. If $xuy \in \Delta^*$, then $xuy \in \Delta^*$.

**Proof.** If $xuy = e$, then $xuy = e$ since $|u| \geq |v|$, and so the conclusion is true. Suppose now that $xuy \in \Delta^*$. Then $xuy = u_1u_2\cdots u_k$, where $u_i \in \Delta$ for each $i, 1 \leq i \leq k$. We consider the following cases.
If \( z \in \Delta^* \), then \( zv \in \Delta^* \), which implies that \( y \in \Delta^* \). Then \( xzy \in \Delta^* \) since \( v \in \Delta^* \).

If \( z \notin \Delta^* \), then \( y \notin \Delta^* \). We have the following result.

**Claim.** \( x = u_1u_2 \cdots u_i, y = u''_i u_{i+1} \cdots u_k \) and \( u_i = u''_i u'''_i \) for some \( i, 1 \leq i \leq k \).

**Proof.** Suppose the claim is not true, i.e., \( u \) is not a factor of \( u_i \) for all \( i \). Then we have \( z = u_1u_2 \cdots u_i \), \( u = u''_i u_{i+1} \cdots u_j \), \( j > i \), \( u''_j u_{j+1} \cdots u_k \) for some \( i, j \) such that \( j > i \), where \( u''_i, u''_j \in \Delta^* \) and \( u''_i, u''_j \in \Delta^+ \) are such that \( u_i = u''_i u'''_i \) and \( u_j = u''_j u'''_j \). Since both \( u_i = u''_i u'''_i \) and \( u = u''_i \cdots u_{j-1} u''_j \) are invertible modulo \( T \), by Proposition 3.1, \( u''_i \) is invertible modulo \( T \), and so \( u''_i = u_i \in \Delta \). Thus, \( u_i = c \) and \( z \in \Delta^* \), a contradiction. \( \Box \)

Let \( < \) be a linear ordering on an alphabet \( A \). We extend the linear ordering to a linear ordering \( < \) on \( A^* \) as

\[
x < y \iff |x| < |y| \text{ or } |x| = |y| \text{ and } x <_{lex} y,
\]

where \( <_{lex} \) denotes the lexicographical ordering on \( A^* \) induced by the given linear order on \( A \).

Using \( \Delta \) and the ordering \( < \), we define a string-rewriting system \( R = R(T) \) on \( A^* \) as

\[
R := \{(u, v) | u, v \in \Delta^* : u \xrightarrow{\Delta} v, \text{ and } u > v \}.
\]  

(3.2)

Since \( < \) is a well-founded ordering on \( A^* \), and since this ordering is compatible with the operation of concatenation, the condition \( u > v \) for each rule \( (u, v) \in R \) immediately implies that the system \( R \) is Noetherian. Furthermore, \( R \) has the following properties.

**Proposition 3.4.** Let \( T \) be a finite, special string-rewriting system on an alphabet \( A \), and let \( R = R(T) \). Then \( R \) is confluent and equivalent to \( T \).

**Proof.** For each rule \( (l, e) \in T, l \in \Delta^* \), so \( (l, e) \in R \). Thus \( T \subseteq R \). On the other hand, for each rule \( (u, v) \in R \), \( u \xrightarrow{T} v \), and so \( \xrightarrow{\Delta} R \subseteq \xrightarrow{T} \). Thus \( R \) is equivalent to \( T \).

To show that \( R \) is confluent, we apply Theorem 2.1. For condition (1), let \( (xy, p), (yz, q) \) be two rules in \( R \). Since \( xz, yz \in \Delta^* \), by Proposition 3.2, \( x, y, z \in \Delta^* \). Thus \( xz, pz \in \Delta^* \). Since \( < \) is a linear ordering, \( xz = pz, xz < pz, \) or \( xz > pz \). By the definition of \( R \), either \( (xz, pz) \) or \( (pz, xz) \) must be a rule of it, or else \( xz = pz \). For condition (2), if \( (xyz, p) \) and \( (y, q) \) are rules in \( R \), then, since \( y > q \), by Proposition 3.3, \( xzq \in \Delta^* \). So either \( (xz, p) \) or \( (p, xzq) \) must be a rule in \( R \), or else \( xzq = p \). \( \Box \)

This technical result will be used in the next section to show that the word problem for a finite, special string-rewriting system is reducible to its restricted version.

### 4. The Word Problem

In this section, we will show that the word problem for finite, special string-rewriting systems is reducible to its restricted version. To this end, we first introduce a normal form theorem for finite, special string-rewriting systems.

**Definition 4.1.** Let \( T \) be a finite string-rewriting system on \( A \), and let \( u, v \in A^* \). \( v \) is a maximal invertible factor of \( u \), if there exist two words \( s, t \in A^* \) such that all of
the following conditions are satisfied:
   (1) \( u = svt \), i.e., \( v \) is a factor of \( u \),
   (2) \( v \) is invertible modulo \( T \), and
   (3) whenever \( s = s_1s_2 \) and \( t = t_1t_2 \) such that \( s_2t_1 \neq e \), then the
       word \( s_2vt_1 \) is not invertible modulo \( T \).

**Theorem 4.2.** (Normal Form Theorem — Otto and Zhang, 1991) Let \( T \) be a finite, special string-rewriting system on \( A \). Then

1. Each \( u \in A^* \) has a unique factorization of the form \( u = u_0a_1u_2 \cdots a_mu_m \) for some nonnegative integer \( m \), where \( u_0, u_1, \ldots, u_m \) are maximal invertible factors of \( u \) (some of which may be empty), and \( a_1, a_2, \ldots, a_m \in A \). The factorization will be called the normal form of \( u \).
2. Let \( u, v \in A^* \), and let \( u = u_0a_1 \cdots a_mu_m \), and \( v = v_0b_1 \cdots b_nv_n \) be the normal forms of \( u \) and \( v \), respectively. Then \( u \twoheadrightarrow r v \) if and only if \( n = m, a_i = b_i \) for each \( i \in \{1, \ldots, m\} \), and \( u_j \twoheadrightarrow r v_j \) for each \( j \in \{0, 1, \ldots, m\} \).

Let \( T \) be a finite, special string-rewriting system on \( A \). For \( x, y \in A^* \), \( x \) and \( y \) are congruent modulo \( T \) (i.e., \( x \equiv_T y \)) if and only if \( y \) is obtained from \( x \) by a sequence of inserting or deleting words of \( \text{dom}(T) \). Theorem 4.2 reflects that deletions and insertions can only occur on the invertible parts of words.

Let \( T \) be a finite, special string-rewriting system on \( A \), and let \( u, v \in A^* \). \( v \) is called the maximal right invertible prefix of \( u \) if \( u = wv \) for some \( w \in A^* \) and \( v \) is right invertible modulo \( T \), and whenever \( w = w_1w_2 \) with \( w_2 \neq e \), then the word \( w_2v \) is not right invertible modulo \( T \).

**Lemma 4.3.** Let \( T \) be a finite, special string-rewriting system on \( A \), and let \( u \in A^* \). Suppose that \( t \) is the maximal right invertible suffix of \( u \) and \( u = wt \) for some \( w \in A^* \). Then \([e]_T = w^{-1}[u]_Tt^{-1}\).

**Proof.** Let \( t \) be the maximal right invertible suffix of \( u \) and let \( u = wt \). Then \([w]_T = [u]_Tt^{-1}\). Hence, it is immediate that we can assume that \( t = e \). Therefore, the conclusion becomes \( u^{-1}[u]_T = [e]_T \). Since \([e]_T \subseteq u^{-1}[u]_T \), we need only to show that \( u^{-1}[u]_T \subseteq [e]_T \).

Suppose \( v \in u^{-1}[u]_T - [e]_T \). Then \( v \neq e \). Since \( R \) is equivalent to \( T \), by Theorem 4.2, if \( u \) has an empty maximal right invertible suffix, so does any descendent modulo \( R \). Without loss of generality, we may assume that both \( u \) and \( v \) are irreducible modulo \( R \). Since \( R \) is confluent and \( u \) is irreducible modulo \( R \), \( uv \) reduces to \( u \) modulo \( R \). But then \( u = u_1u_2, v = v_1v_2 \), where \( u_1, u_2, v_1, v_2 \in A^* \) are such that \( u_2v_1 \in \Delta^* \) and \( u_2 \neq e \neq v_1 \). Hence, \( u_2 \) is a nonempty right invertible suffix of \( u \). This is a contradiction. \( \square \)

**Lemma 4.4.** Let \( T = \{(l_1, e), (l_2, e), \ldots, (l_k, e)\} \) be a finite, special string-rewriting system on \( A \). If \([e]_T \) is recursive, then the factorization of each \( l_i \) in (1) in Section 3 is effectively computable. Further, the computation requires a linear number of queries of type \( x \) in \([e]_T \).

**Proof.** Let \( l_i = a_1a_2 \cdots a_n \), where \( a_i \in A \). Then \( l_i = a_1a_2 \cdots a_n \twoheadrightarrow r e \), and so a nonempty prefix \( a_1a_2 \cdots a_m \) of \( l_i \) is invertible modulo \( T \) if and only if \( a_{m+1} \cdots a_n a_1a_2 \cdots a_m \twoheadrightarrow r e \), i.e., \( a_{m+1} \cdots a_n a_1a_2 \cdots a_m \in [e]_T \). So, by the definition of minimal words, \( a_1a_2 \cdots a_p \) is a minimal word if and only if \( a_{p+1} \cdots a_n a_1a_2 \cdots a_p \in [e]_T \) and \( a_{q+1} \cdots a_n a_1a_2 \cdots a_q \notin [e]_T \).
[e]_T for each q < p. Since [e]_T is recursive, we can find the first minimal factor l_{i,1} of l_i by testing whether a_{p+1} \cdots a_k a_1 a_2 \cdots a_p \in [e]_T for p < n. Let l_{i,1} = a_1 a_2 \cdots a_p. If p = n, we are done. Otherwise, since a_{p+1} \cdots a_k l_{i,1} \not\rightarrow_T e, we can find the second minimal factor of l_i by dealing with a_{p+1} \cdots a_k l_{i,1} in a similar way. Repeating this procedure, we can finally find the factorisation of l_i in (1).

It can easily be seen that the above computation takes a linear number of queries of type x in [e]_T.

Lemma 4.5. Let T be defined as in Lemma 4.4 and let the sets \Delta_{i,p} and \Delta be defined as in Section 3. If [e]_T is recursive, then all these sets are effectively computable.

Proof. Let \Delta_{i,p} consist of the minimal words which are congruent to the minimal factor l_{i,p} of l_i \in dom(T) in (1). Since l_i = l_{i,1} \cdots l_{i,p} l_{i,p+1} \cdots l_{i,n}, and all l_{i,j} are invertible modulo T, we have l_{i,p} \cdots l_{i,n} l_{i,1} \cdots l_{i,p-1} \not\rightarrow_T e. So, for each word w \in A^*, w \not\rightarrow_T l_{i,p} if and only if w l_{i,p+1} \cdots l_{i,n} l_{i,1} \cdots l_{i,p-1} \not\rightarrow_T e, i.e., w l_{i,p+1} \cdots l_{i,n} l_{i,1} \cdots l_{i,p-1} \in [e]_T. Hence, for a word w \in A^* with \|w\| \leq \max_{1 \leq i \leq k} |l_i|, whether w is in \Delta_{i,p} can be determined by just testing whether w l_{i,p+1} \cdots l_{i,n} l_{i,1} \cdots l_{i,p-1} \in [e]_T holds. Thus, \Delta_{i,p} is effectively computable. Since \Delta is the union of all \Delta_{i,p}, it is effectively computable.

Theorem 4.6. Let T be a finite, special string-rewriting system on A. Then the following three assertions are effectively equivalent.

1) The word problem for T is decidable.
2) The congruence class [x]_T is recursive for any fixed x \in A^*.
3) The congruence class [e]_T is recursive.

Proof. 1) \Rightarrow 2). It is trivial.
2) \Rightarrow 3). Let z = a_1 a_2 \cdots a_k. It follows directly from the following equality (Lemma 4.3):

\[ [e]_T = [z]_T z^{-1} \cap a_1^{-1} [z]_T (a_2 \cdots a_k)^{-1} \cdots (a_1 \cdots a_{k-1})^{-1} [z]_T a_k^{-1} \cap z^{-1} [z]_T. \]

3) \Rightarrow 1). Let R = R(T) be defined as in Section 3. Then R is Noetherian, confluent and equivalent to T. Let [e]_T be recursive. We have the following result.

Claim. Let \Delta be defined as in Section 3, and let x, y \in \Delta^*. Then, whether x \not\rightarrow_T y is decidable.

Proof. Since [e]_T is recursive, by Lemma 4.5, the sets \Delta_{i,p} are effectively computable. For each i \leq k and p \leq n_i, l_{i,p} \in \Delta_{i,p}. Since the factorisation of each l_i into its minimal factors is effectively computable (Lemma 4.4), we can find its inverse l'_{i,p} \in A^* for each l_{i,p}, i.e., l'_{i,p} l_{i,p} \not\rightarrow_T l_{i,p} l'_{i,p} \not\rightarrow_T e. Let z = x_1 x_2 \cdots x_s \in \Delta^*, where x_i \in \Delta_{j_i,m_i}, 1 \leq i \leq s. Then z = x_1 x_2 \cdots x_s \not\rightarrow_T l_{j_1,m_1} l_{j_2,m_2} \cdots l_{j_s,m_s}. So, z \not\rightarrow_T y if and only if y \not\rightarrow_T l_{j_1,m_1} l_{j_2,m_2} \cdots l_{j_s,m_s} if and only if y l'_{j_1,m_1} l'_{j_2,m_2} \cdots l'_{j_s,m_s} \not\rightarrow_T e, i.e., y l'_{j_1,m_1} l'_{j_2,m_2} \cdots l'_{j_s,m_s} \in [e]_T, which is decidable.

Since R is Noetherian and confluent, by the above claim, whether a word w is reducible modulo R = R(T) is decidable and if it is then the irreducible descendant is effectively computable. Therefore, the word problem for T is decidable.

Following Theorem 4.6, we get
THEOREM 4.7. Let $T$ be a finite, special string-rewriting system on $A$ and let $z \in A^*$. Then the word problem for $T$ is reducible to its restricted version:

- **Instance:** A word $w \in A^*$;
- **Question:** Is $w$ is congruent to $z$ modulo $T$?

Furthermore, the following result is implicitly in Adjan (1966) and Makanin (1966): the word problem for a finite, special string-rewriting system $T$ is equivalent to the word problem of the group of units of the monoid $M(A; T)$. Using this result, we can also prove the equivalence of 1) and 3) in Theorem 4.6.

5. A Markov Type Theorem

Let $\mathcal{M}$ be a property of finitely presented (special) monoids which is preserved under isomorphism. The property $\mathcal{M}$ is said to be a Markov property if

1. there is a finitely presented (special) monoid $M_1$ which can not be embedded in any finitely presented (special) monoid with $\mathcal{M}$, and
2. there is a finitely presented (special) monoid $M_2$ with $\mathcal{M}$.

It is well known that the Markov property is undecidable for finitely presented monoids (Markov, 1951). In the case of groups, a similar result holds (Rabin, 1958).

In this section, we will prove that a Markov type property is undecidable for finite, special string-rewriting systems. To this end, we need the following results.

Let $T$ be a finite, special string-rewriting system on $A$ such that all letters $a \in A$ are not congruent to $e$ modulo $T$ and are not pairwise congruent modulo $T$. Further, let $s$ and $t$ be two additional letters, and let $B = A \cup \{s, t\}$. Obviously, $T$ can be considered as a string-rewriting system on $B$. Now, for a word $u \in A^*$, let $T_u = T \cup \{(sut, e), (stt, e), (sat, e) | a \in A\}$.

**Proposition 5.1.** If $u \not\sim_T e$, then the special monoid $M(B; T_u)$ is trivial.

**Proof.** Let $u \not\sim_T e$. Then, $st \not\sim_T u$, $stt \not\sim_T u$, $sat \not\sim_T u$. Furthermore, for any $a \in A$, $a \not\sim_T u$, $sat \not\sim_T u$. Hence $M(B; T_u)$ is trivial. $\Box$

Now, we turn to the case in which $u$ is not congruent to $e$ modulo $T$. Note that $T$ is equivalent to a Noetherian and confluent system $T'$ (with respect to the lexical ordering). $T'$ may be infinite and not effective, but this is irrelevant. For $x \in A^*$, let $x' \in IRR(T')$ with $[x]_T = [x']$. Since all letters $a$ in $A$ are not congruent to the identity and are pairwise not congruent modulo $T$, we have $a = a'$. Thus $T_u$ is equivalent to

$$T'_u = T' \cup \{(su't, e), (stt, e), (sat, e) | a \in A\}.$$ 

Obviously, $T'_u$ is Noetherian. Since $u'$ is not congruent to $e$, there exists no new overlappings between the rules of $T'_u$, and so $T'_u$ is confluent. Furthermore, $IRR(T'_u) \subseteq IRR(T_u)$. This leads to the following result.

**Proposition 5.2.** If $u \not\sim_T e$, then $M(A; T)$ is embedded in $M(B; T_u)$.
Now we can prove our main theorem in this section, which is formulated according to an anonymous referee’s suggestion.

**Theorem 5.3.** Let $P$ be a property of finite, special string-rewriting systems satisfying the following two conditions:

1. $P$ implies a given fixed Markov property $M$ of finitely presented special monoids, i.e., for a finite, special string-rewriting system $T$ on $A$, if $T$ has $P$, then $M(A; T)$ has $M$;
2. there exists a finite, special string-rewriting system $R$ on $C$ such that, for any finite, special string-rewriting system $T$ on $A$, $T$ has $P$ whenever $M(A; T) \preceq M(C; R)$.

Then the property $P$ is undecidable for finite, special string-rewriting systems.

Theorem 5.3 generalizes the following result due to O’Dunlaing (1983): a property $P$ is undecidable for finite string-rewriting systems if it satisfies (1) $P$ is invariant under the equivalence of string-rewriting systems, (2) every trivial string-rewriting system has the property $P$, and (3) every string-rewriting system in $P$ has a decidable word problem.

**Proof.** Let $P$ be a property desired for finite, special string-rewriting systems. Then the condition (1) implies that there exists a finite, special string-rewriting system $T_1$ on alphabet $A_1$ such that the monoid $M(A_1; T_1)$ can not be embedded in any special monoid $M(B; R)$ presented by a finite, special string-rewriting system $R$ on an alphabet $B$ that has property $P$. In addition, let $T_3$ be a finite, special string-rewriting system on $A_3$ with the ‘universal’ property described in condition (2), and let $T_3$ be a finite, special string-rewriting system on $A_3$ with undecidable word problem. Without loss of generality, we may assume that the alphabets $A_1, A_2, A_3$ are pairwise disjoint, and that all letters in $A_i$ are not congruent to the identity $e$ modulo $T_i$ and are pairwise not congruent modulo $T_i$.

Now we consider the system $T = T_1 \cup T_3$ on alphabet $A = A_1 \cup A_3$. Then $M(A_1; T_1)$ and $M(A_3, T_3)$ are embedded in $M(A; T)$. Hence $T$ does not have the property $P$ due to the choice of $T_1$, and its word problem is undecidable due to the choice of $T_3$. So by Theorem 4.7, whether a word is congruent to the identity $e$ is undecidable. On the other hand, the restricted version of the word problem for $T$ is effectively reducible to the problem of deciding the property $P$ as we will see in the following. Hence, the latter problem is in fact undecidable.

Let $s$ and $t$ be two new letters, which are not contained in $A_1 \cup A_2 \cup A_3$, and let $B = A \cup \{s, t\}$. We will describe an effective process that, given a word $u \in A^*$, yields a finite, special string-rewriting system $T_4$ on $A_4$ satisfying the equivalence:

$$(*): T_4 \text{ has the property } P \text{ if and only if } u \xrightarrow{\ast} e. $$

So let $u \in A^*$. Using the construction before Proposition 5.1, we obtain a finite, special string-rewriting system $T_u$ on alphabet $B$ such that either $u \xrightarrow{\ast} T e$ and the monoid $M(B; T_u)$ is trivial, or $u \xrightarrow{\not\ast} T e$ and the monoid $M(A; T)$ is embedded in $M(B; T_u)$. Let $A_4 = B \cup A_2$, and $T_4 = T_u \cup T_2$. Then, the system $T_4$ on $A_4$ can be constructed effectively from $u$, since $T$ and $T_2$ are given in advance. It remains to be verified that the special system $T_4$ on $A_4$ does indeed satisfy the equivalence ($\ast$). So assume first that $u \xrightarrow{\ast} T e$, then the special monoid $M(B; T_u)$ is trivial, and hence, $M(A_4; T_4) \cong M(B; T_u) \ast M(A_2; T_2) \cong M(A_2; T_2)$. Now, by the hypothesis on $T_2$, $T_4$ does have $P$. 


On the other hand, if we have \( u \xrightarrow{\gamma} e \), Proposition 5.2 yields the following chain of embeddings \( M(A_1; T_1) \to M(A; T) \to M(B; T_u) \to M(A_4; T_4) \). Hence by the choice of \( T_1, T_4 \) does not have \( P \). Thus the equivalence \( (*) \) is satisfied, i.e., the restricted version of the word problem for \( M(A; T) \) is indeed effectively reducible to the problem of deciding \( P \). This completes the proof of Theorem 5.3. □

Obviously, Theorem 5.3 implies that Markov properties are undecidable for finitely presented special monoids. Since the property of being a group is a Markov Property for finitely presented special monoids, we get the following results.

**Corollary 5.4.** (Narendran et al, 1991) *It is undecidable in general whether a finitely presented special monoid is a group.*

However, it is decidable whether a special monoid presented by a one-rule string-rewriting system is a group (Adjan, 1966).

On the other hand, from Theorem 5.3, we obtain that some computational properties are also undecidable.

**Proposition 5.5.** (1) *It is undecidable whether the Knuth-Bendix completion algorithm terminates on a finite, special string-rewriting system.*

(2) *It is undecidable whether a finite, special string-rewriting system is equivalent to a finite, Noetherian string-rewriting system that is confluent on a given congruence class.*

**Proof.** (1). Let \( P \) denote the property that the Knuth-Bendix completion algorithm terminates on \( T \) of finite, special string rewriting systems \( T \). We need only to verify that \( P \) satisfies the two conditions in Theorem 5.3. Actually, for a finite, special string-rewriting system \( T \), if \( T \) has \( P \), then the word problem is decidable for the monoid \( M(A; T) \), which is a Markov property of finitely presented special monoids. Furthermore, if a finite, special string-rewriting \( T \) presents a trivial monoid, then the Knuth-Bendix completion algorithm terminates on \( T \). Hence (1) holds.

Similarly, (2) holds. □

**Acknowledgment**

Special thanks are due to Professor Dr. Friedrich Otto for many comments on the first draft of this paper, especially for pointing out an error in Lemma 4.3, and also to Dr. Richard Atkins and the referees for their helpful suggestions in revising this paper.
References


