Juhani Karhumäki | Michal Kunc | Alexander Okhotin

Computing by Commuting

Turku Centre *for* Computer Science

TUCS Technical Report
No 733, December 2005

Computing by Commuting

Juhani Karhumäki
Department of Mathematics, University of Turku and
Turku Centre for Computer Science
FIN–20014 Turku, Finland
karhumak@utu.fi

Michal Kunc
Department of Mathematics, University of Turku and
Turku Centre for Computer Science
FIN–20014 Turku, Finland
kunc@math.muni.cz

Alexander Okhotin
Department of Mathematics, University of Turku and
Turku Centre for Computer Science
FIN–20014 Turku, Finland
aleokh@utu.fi

TUCS Technical Report

No 733, December 2005

Abstract

We consider the power of the following rewriting: given a finite or regular set X of words and an initial word w, apply iteratively the operation which deletes a word from X from one of the ends of w and simultaneously catenates another word from X to the opposite end of w. We show that if the deletion is always done at the beginning and the catenation at the end, and the choice of a word to be catenated does not depend on the word erased, then the generated language is always regular, though the derivability relation is not, in general, rational. If the deletion and the catenation are done arbitrarily at the opposite ends, the language need not be regular. If the catenation is done at the same end as the deletion, the relation of derivability is rational even if the catenated word can depend on the word erased.

Keywords: Rewriting systems, regular languages, commutation of languages, rational relations.

TUCS Laboratory
Discrete Mathematics for Information Technology

1 Introduction

Operations on words, and the properties preserved under these operations, have always been and still are an essential part of formal language theory. In particular, operations preserving nice properties, such as the regularity, have been searched for. Recently a number of such operations were motivated by two active research topics. On one hand, the formalization of DNA computing, see [22], has introduced a rich variety of new operations on words. On the other hand, Conway's problem on commutation of languages, see [17, 13], has motivated to reconsider some similarly formulated operations on words.

We consider the following basic operation, which we call *one-way rewriting*, and its natural variants and extensions. Let w be a finite word and X a finite set of words. For each $x, y \in X$ we rewrite w to $(x^{-1}w)y$, that is, we first delete x from the beginning of w and then add y to the right end of $x^{-1}w$. An obvious question is: Is the language consisting of all words obtained in this way necessarily regular? We prove that the language is indeed always regular, though the rewriting itself is not a rational relation.

It is important to note that both deletions and additions are done freely in the sense that there is no connection between words x and y used at a step of rewriting. If for each prefix being erased the choice of the word to be appended is predetermined, we obtain the well-known tag systems of Post [25], which are, in general, powerful enough to simulate Turing machines [19]. We shall prove that if we impose even a single such constraint, i.e., when deleting a certain $x \in X$ we have to add a particular $x' \in X$, we can generate nonregular languages.

There are several variants of the above rewriting, and some simpler cases have been studied before. Following Post [25], Büchi [5] proved the regularity of the set of words derivable from a single word using a finite set of rules $x \to y$ that specify local rewriting at the beginning of the word, that is, xw is rewritten with yw (*one-sided rewriting*). The regularity is still preserved by one-sided rewriting with any recognizable set of rules, which follows from Conway's result on a certain larger class of language transformations [9, Th.10]. Kratko [15] and, independently, Büchi and Hosken [6] extended the result of Büchi [5] in another direction, showing that if the rewriting is done locally at both of the ends of a word (*two-sided rewriting*) using a finite set of rules, then the generated language is regular as well. In addition, one-sided rewriting was proved to be a rational relation by Caucal [7]. We generalize these results by proving that the derivability relation of two-sided rewriting is rational for any recognizable set of rules.

Another variant is obtained by allowing *two-way rewriting*: the deletion can be done at either of the ends, but the simultaneous adding has to be done at the opposite end. More formally, if $w = xw'$, then w derives $w'y$, and if $w = w'x$, then w derives yw'; in other words, w is rewritten to $(x^{-1}w)y$ or $y(wx^{-1})$. We show that nonregular languages can be generated, and that

their recursiveness in general appears to be a challenging problem.

The above two-way rewriting is a natural sequential variant of commutation of languages, $KL = LK$ [13, 14]. Similarly, one-way rewriting is a sequential variant of so-called semi-commutation of languages, $KL \subseteq LK$. There are remarkable recent results on these language problems, such as the negative answer to Conway's problem, that is, that the greatest set commuting with a given regular (or even finite) set need not be recursively enumerable [17], while in the case of semi-commutation it is always regular [16]. Our paper is motivated by these results — up to the point of its title. On the other hand, it is intended to be the first systematic study of the corresponding sequential problems.

In our presentation we need only basic results and notions of formal language and automata theory that can be found in the standard textbooks [28, 11], as well as some background in rational relations [3].

2 Two-sided local rewriting

Let us start from the simplest mode of rewriting, where, at every step, either a prefix is replaced by another word, or a suffix is replaced by another word. From the work of Kratko [15] and Büchi and Hosken [6] it is known that for any finite set of rules of this form the set of words derivable from a given initial word is regular. We shall first extend this result to the case of an infinite recognizable set of rewriting rules and any fixed regular set of initial words. Then we shall prove a stronger result that every such rewriting system implements a rational relation.

Let Σ be an alphabet and let $I \subseteq \Sigma^*$ be a regular set of initial words. Let the set of rewriting rules $x \xrightarrow{\ell} y$ admissible at the left end be a recognizable relation (in other words, regular as a language over $\Sigma \cup \{\xrightarrow{\ell}\}$); then it can be written in the form $\xrightarrow{\ell} = \bigcup_{i=1}^{n} X_i \times Y_i$, where $X_i, Y_i \subseteq \Sigma^*$ are regular languages. Similarly, let the set of rules $u \xrightarrow{r} v$ at the right end be recognizable and denote it as $\xrightarrow{r} = \bigcup_{i=1}^{n} U_i \times V_i$, where $U_i, V_i \subseteq \Sigma^*$ are regular.

Define the binary relation of *one-step derivability* on the set Σ^*:

1. For every rule $x \xrightarrow{\ell} y$ and for every $w \in \Sigma^*$, $xw \Longrightarrow yw$.
2. For every rule $u \xrightarrow{r} v$ and for every $w \in \Sigma^*$, $wu \Longrightarrow wv$.

The reflexive and transitive closure of this relation, denoted by $\Longrightarrow^*$, is the relation of *derivability*. The language generated by the rewriting system is the set of words derivable from some word in I.

In order to prove that $\Longrightarrow^*$ is a rational relation, we first need to show that the language derivable via $\Longrightarrow$ from any fixed regular set I is also regular.

Theorem 2.1. *The language generated by any two-sided local rewriting system is regular, and, given finite automata for I, $\overset{\ell}{\to}$ and $\overset{r}{\to}$, a finite automaton for this language can be effectively constructed.*

The proof proceeds as follows. First we represent deletion of a symbol a as a concatenation of a "negative" symbol: $\overrightarrow{a}$ if it is erased from the left and $\overleftarrow{a}$ if it is erased from the right. The resulting set of "computation histories" of our rewriting is regular. Second, we prove that every such history of derivation of a word is equivalent to this word itself, under a congruence defined by $\overrightarrow{a}a = a\overleftarrow{a} = \varepsilon$ for all $a \in \Sigma$. Then we consider this congruence as an operation on languages; by analogy to the well-known results on reduction in free groups, see e.g. [2, 23], it can be shown that it preserves regularity, and so the language generated by the rewriting system must be regular.

Let us proceed with the proof. Define two copies of the alphabet Σ, $\overrightarrow{\Sigma} = \{\overrightarrow{a} \mid a \in \Sigma\}$ and $\overleftarrow{\Sigma} = \{\overleftarrow{a} \mid a \in \Sigma\}$. For every word $w = a_1 \dots a_\ell \in \Sigma^*$, with $\ell \geqslant 0$, denote $\overrightarrow{w} = \overrightarrow{a}_\ell \dots \overrightarrow{a}_1$ and $\overleftarrow{w} = \overleftarrow{a}_\ell \dots \overleftarrow{a}_1$; note that besides marking the symbols we also reverse their order. This notation is extended to languages $L \subseteq \Sigma^*$ as $\overrightarrow{L} = \{\overrightarrow{w} \mid w \in L\}$ and $\overleftarrow{L} = \{\overleftarrow{w} \mid w \in L\}$. Define the alphabet $\Sigma_3 = \Sigma \cup \overrightarrow{\Sigma} \cup \overleftarrow{\Sigma}$, and consider the following reduction rules on Σ_3^*: $\overrightarrow{a}a \to \varepsilon$ and $a\overleftarrow{a} \to \varepsilon$, for all $a \in \Sigma$. A word $\alpha \in \Sigma_3^*$ is said to be *reducible* to $\beta \in \Sigma_3^*$ if and only if it can be transformed to β by zero or more such reductions.

We can now represent our rewriting system in terms of this transformation. Define the following regular language over Σ_3:

$$L_0 = \Big(\bigcup_{i=1}^{n} Y_i \overrightarrow{X}_i\Big)^* \cdot I \cdot \Big(\bigcup_{i=1}^{n} \overleftarrow{U}_i V_i\Big)^*$$

Lemma 2.2. *A word $w \in \Sigma^*$ is derivable in the two-sided rewriting system if and only if there exists $\alpha \in L_0$ reducible to w.*

Proof. First we assume that w is derivable in zero or more steps, and show, by induction on the length of the derivation of w, that there exists $\alpha \in L_0$ reducible to w.

Basis: If the derivation is of length 0, then $w \in I$ and hence $w \in L_0$.

Induction step: Suppose w is derivable in one or more steps from some $w_0 \in I$, and consider the last step in the derivation, which can be assumed, without loss of generality, to be a rewriting at the left end: $w_0 \Longrightarrow \dots \Longrightarrow xw_1 \Longrightarrow yw_1 = w$ using a rule $x \overset{\ell}{\to} y$. Since xw_1 is derived in one step less than w, by the induction hypothesis, xw_1 can be obtained by reducing some $\alpha \in L_0$. Consider the word $y\overrightarrow{x}\alpha$, and reduce it to $y\overrightarrow{x}xw_1$ by exactly the same sequence of cancellations, and then further to yw_1 using $|x|$ more cancellations. Since $\alpha \in L_0$ implies $y\overrightarrow{x}\alpha \in L_0$, it has been shown that some word in L_0 can be reduced to $w = yw_1$.

Second, suppose a word $w \in \Sigma^*$ can be obtained by reducing a word

$$\alpha = y_m \overrightarrow{x}_m \dots y_1 \overrightarrow{x}_1 w_0 \overleftarrow{u}_1 v_1 \dots \overleftarrow{u}_k v_k \in L_0,$$

where $x_i \xrightarrow{\ell} y_i$ for all $i \leqslant m$, $u_i \xrightarrow{r} v_i$ for all $i \leqslant k$ and $w_0 \in I$. We prove, by induction on $m + k$, that w can be derived in the rewriting system.

Basis: If $m + k = 0$, then $w = w_0 \in I$.

Induction step: Let $m + k \geqslant 1$. Let us first assume that the last reduction rule that was applied is of the form $\overrightarrow{a}a \to \varepsilon$. Then, in every intermediate word in the reduction of α to w there was an occurrence of $\overrightarrow{a}$ between the prefix y_m and every occurrence of letters from $\overleftarrow{\Sigma}$. Because letters of the prefix y_m can only be removed by a rule of the form $c\overleftarrow{c} \to \varepsilon$, this means that y_m is never touched.

We would like to know that all deletions removing letters of the factor $\overrightarrow{x}_m$ are performed at the end of the reduction. Assume that it is not the case. Then there is a step of the form $y_m \overrightarrow{z}\,\overrightarrow{c}\,c\gamma \to y_m \overrightarrow{z}\gamma$, where $z \in \Sigma^*$, $c \in \Sigma$, $\gamma \in \Sigma_3^*$ and $\overrightarrow{z}\,\overrightarrow{c}$ is the remaining prefix of $\overrightarrow{x}_m$, such that right after this step a reduction not involving letters of $\overrightarrow{x}_m$ is performed. Because the latter reduction has to occur somewhere inside the suffix γ, the order of these two reductions can be exchanged. Repeating this argument several times, we eventually move the whole deletion of $\overrightarrow{x}_m$ to the end, and so the word $\alpha = y_m \overrightarrow{x}_m \beta$ is first reduced to $y_m \overrightarrow{x}_m x_m w'$, where $w' \in \Sigma^*$, without touching the prefix $y_m \overrightarrow{x}_m$. Next, the word obtained is reduced to $w = y_m w'$.

Note that $\beta \in L_0$, and that we can reduce β to $x_m w'$ following the same steps as in the reduction of $y_m \overrightarrow{x}_m \beta$ to $y_m \overrightarrow{x}_m x_m w'$. Since the sum $m + k$ for $\beta \in L_0$ is less by one than the corresponding sum for α, we can apply the induction hypothesis to obtain that $x_m w'$ is derivable in the rewriting system. Since $x_m w' \Longrightarrow y_m w'$ by the rule $x_m \xrightarrow{\ell} y_m$, the word $y_m w' = w$ is derivable as well.

If the last reduction uses a rule of the form $a\overleftarrow{a} \to \varepsilon$, the proof can be done symmetrically. □

In light of Lemma 2.2, it is enough to show that for any regular set $L \subseteq \Sigma_3^*$, the set of words obtained by reduction from words in L is regular as well. This is very similar to the known problem of reduction in free groups, that is, given a language L over an alphabet $\Sigma \cup \Sigma'$, where $\Sigma' = \{a' \mid a \in \Sigma\}$, and an equivalence $aa' = a'a = \varepsilon$ for all $a \in \Sigma$, determine the language of words derivable by reduction from some element of L. This transformation preserves regularity, which was first established by Benois [2], see also Berstel [3, p. 59], an alternative proof by Pin and Sakarovitch [23], and a detailed treatment by Sakarovitch [27, Ch. II, Sec. 6]. In our case, we have to make a distinction between right- and left-sided cancellation, but such cases, as pointed out by Pin and Sakarovitch [23], can be treated in the same fashion. For completeness, we give a simple proof of this result in the following lemma, which is actually a special case of a more general theorem

of Book and Otto [4, Th. 4.1.2], as well as a special case of a related general result due to Hofbauer and Waldmann [12].

Lemma 2.3. *For every finite automaton A over Σ_3, the set of all words obtained from some element of $L(A)$ by reduction is regular. A finite automaton accepting this set can be effectively constructed.*

Proof. Let $D \subseteq \Sigma_3^*$ be the set of all words reducible to the empty word. This is a variant of the Dyck language defined by a one-nonterminal context-free grammar with the rules $S \to \overrightarrow{a}Sa$, $S \to aS\overleftarrow{a}$, $S \to SS$ and $S \to \varepsilon$. It is easy to see that every reduction via the rules $\overrightarrow{a}a \to \varepsilon$ and $a\overleftarrow{a} \to \varepsilon$ is equivalent to removing several factors belonging to D. Such a removal can be formalized by a regularity-preserving inverse substitution as follows.

Let e be a new symbol not in Σ_3 and consider the context-free substitution f from $(\Sigma_3 \cup \{e\})^*$ to Σ_3^* defined by the rule $f(e) = D$ and identical otherwise. Let $h\colon (\Sigma_3 \cup \{e\})^* \to \Sigma_3^*$ be the morphism sending e to the empty word and leaving other symbols unchanged. The set of words over Σ_3 that can be obtained from words in $L(A)$ by reduction can now be represented as $h(f^{-1}(L(A)))$.

It is well-known that regular languages are effectively closed under morphisms. Inverses of arbitrary substitutions also preserve regularity [27, Ch. II, Cor. 3.18], and this closure is effective if for the languages being substituted one can decide the emptiness of intersection with a given regular language. The decidability of this property for context-free languages is known, and hence our inverse Dyck substitution f^{-1} effectively preserves regularity. Thus we can deduce that the language $h(f^{-1}(L(A)))$ is regular and an automaton recognizing it can be algorithmically constructed. □

Now, the proof of Theorem 2.1 follows from these lemmata. On one hand, the set of all words over Σ^* obtained by reduction of some word from L_0 equals the language generated by our rewriting system, according to Lemma 2.2. On the other hand, it is effectively regular by Lemma 2.3.

Theorem 2.1 states that the derivability relation $\Longrightarrow^*$ preserves regularity. Next we are going to use this result to prove a stronger statement saying that this relation is, in fact, rational.

Theorem 2.4. *For any recognizable relations $\overset{\ell}{\to}$ and $\overset{r}{\to}$, the corresponding derivability relation $\Longrightarrow^*$ is rational, and a rational expression for it can be effectively constructed.*

Proof. Let $K \cdot L^{-1} = \{u \mid \exists v \in L : uv \in K\}$ denote the quotient of languages. Let us assume that $\varepsilon \overset{\ell}{\to} \varepsilon$ and $\varepsilon \overset{r}{\to} \varepsilon$; this assumption does not change the generative power of the rewriting and allows us to consider only those derivations where some rule was applied on each side.

We will split the derivation relation into several rational relations according to whether the rewritings performed on the left and on the right interfere

or not, and which rewriting rules are used. To define these relations, we have to introduce several auxiliary languages.

Let K_i^ℓ (resp. K_k^r) be the language consisting of all words derivable from the language Y_i (resp. V_k) using only the rules from $\xrightarrow{\ell}$ (resp. $\xrightarrow{r}$); it is regular by Theorem 2.1 applied to a rewriting system with the initial set Y_i (resp. V_k) and the given set of one-sided rules. Further, let N_i^ℓ (resp. N_k^r) be the language consisting of all words from which one of the words of the language X_i (resp. U_k) can be derived using only the rules from $\xrightarrow{\ell}$ (resp. $\xrightarrow{r}$). The regularity of N_i^ℓ and N_k^r follows by an application of Theorem 2.1 to another one-sided rewriting system, in which the set of rewriting rules is the inverse of $\xrightarrow{\ell}$ (resp. $\xrightarrow{r}$).

For $i, k \in \{1, \ldots, n\}$, let us define a rational relation

$$\rho_{ik} = (N_i^\ell \times K_i^\ell) \cdot \{(w, w) \mid w \in \Sigma^*\} \cdot (N_k^r \times K_k^r).$$

As we will verify, this relation consists of all derivations where a certain part of the initial word remains unchanged and rewriting on both sides is performed independently.

In order to define languages required to deal with the case when rewriting on the left eventually removes some letters previously added on the right, let us consider arbitrary $i, j, k \in \{1, \ldots, n\}$, where i and k denote the indices of the sets $X_i \times Y_i$ and $U_k \times V_k$ containing the rules applied during the right-most deletion on the left and the left-most deletion on the right, respectively, before the two rewritings interfere during an application of a rule from the set $X_j \times Y_j$. We have to classify all factors w remaining from the original word, which can be removed in this step of the derivation, into finitely many languages L_{ijq}^ℓ, according to which words $\hat{w}$ composed of letters previously added on the right can be appended to get a word from X_j. Equivalently, such words $w\hat{w}$ are required to belong to the language $(K_i^\ell)^{-1} \cdot X_j$, and therefore this classification can be done using an automaton recognizing this language. So let $A_{ij}^\ell = (\Sigma, Q_{ij}^\ell, q_0, \delta_{ij}^\ell, F_{ij}^\ell)$ be a DFA recognizing the language $(K_i^\ell)^{-1} \cdot X_j$ and for every $q \in Q_{ij}^\ell$ consider the language $L_{ijq}^\ell = \{w \in \Sigma^* \mid \delta_{ij}^\ell(q_0, w) = q\}$. Then, for every $q \in Q_{ij}^\ell$, denote by M_{ijkq}^ℓ the set of all words derivable using $\Longrightarrow^*$ from $Y_j \cdot (((K_i^\ell \cdot L_{ijq}^\ell)^{-1} \cdot X_j)^{-1} \cdot K_k^r)$, which is a regular language by Theorem 2.1. Finally, define a recognizable relation $\sigma_{ijkq}^\ell = N_i^\ell L_{ijq}^\ell N_k^r \times M_{ijkq}^\ell$.

Relations σ_{ijkq}^r, corresponding to derivations where the remaining factor of the original word is removed from the right, are defined symmetrically. As usual, we denote the reversal of a word $w = a_1 \ldots a_m$ by $w^R = a_m \ldots a_1$ and extend the notation to languages as $L^R = \{w^R \mid w \in L\}$. This time we take a DFA $A_{jk}^r = (\Sigma, Q_{jk}^r, q_0, \delta_{jk}^r, F_{jk}^r)$ for the language $(U_j \cdot (K_k^r)^{-1})^R$ and for all $q \in Q_{jk}^r$ we define $L_{jkq}^r = \{w \in \Sigma^* \mid \delta_{jk}^r(q_0, w^R) = q\}$. Further, for every $q \in Q_{jk}^r$, let M_{ijkq}^r be the language of words derivable from $(K_i^\ell \cdot (U_j \cdot (L_{jkq}^r \cdot K_k^r)^{-1})^{-1}) \cdot V_j$ and define the relation $\sigma_{ijkq}^r = N_i^\ell L_{jkq}^r N_k^r \times M_{ijkq}^r$.

Now we define a rational relation

$$\rho = \bigcup_{i,k=1}^{n} \rho_{ik} \cup \bigcup_{i,j,k=1}^{n} \bigcup_{q \in Q_{ij}^{\ell}} \sigma_{ijkq}^{\ell} \cup \bigcup_{i,j,k=1}^{n} \bigcup_{q \in Q_{jk}^{r}} \sigma_{ijkq}^{r}.$$

Notice that this relation can be effectively constructed due to Theorem 2.1. We are going to prove that ρ is equal to $\Longrightarrow^*$.

Let us consider any words w_1 and w_2 such that $w_1 \Longrightarrow^* w_2$, and prove that the pair (w_1, w_2) belongs to ρ. We have to distinguish two cases.

First, we assume that during the derivation of w_2 no letter added by rewriting on one side is removed when rewriting on the other side. Then (after possibly interchanging the order of independent rewritings on different sides) the derivation is of the form

$$w_1 = w_1^{\ell} w_3 w_1^{r} \Longrightarrow^* x_i w_3 u_k \Longrightarrow^2 y_i w_3 v_k \Longrightarrow^* w_2^{\ell} w_3 w_2^{r} = w_2, \tag{1}$$

where $w_3 \in \Sigma^*$ is the common factor of w_1 and w_2 consisting of all those letters that are never removed during the derivation, and $x_i \in X_i$, $y_i \in Y_i$, $u_k \in U_k$ and $v_k \in V_k$, where i and k are the indices of the right-most derivation on the left and of the left-most derivation on the right, respectively. Then it is clear that $w_1^{\ell} \in N_i^{\ell}$, $w_2^{\ell} \in K_i^{\ell}$, $w_1^{r} \in N_k^{r}$ and $w_2^{r} \in K_k^{r}$, showing that $(w_1, w_2) \in \rho_{ik}$.

Second, we consider the derivation where some letter produced by rewriting on the right is later removed by rewriting on the left, as illustrated in Figure 1. The actual derivation is of the form

$$w_1 = w_1^{\ell} w_3 w_1^{r} \Longrightarrow^* x_i w_3 u_k \Longrightarrow^2 y_i w_3 v_k \Longrightarrow^* \bar{w}^{\ell} w_3 \bar{w}^{r} = x_j \tilde{w} \Longrightarrow y_j \tilde{w} \Longrightarrow^* w_2,$$

where $w_3 \in \Sigma^*$ is the factor of w_1 consisting of all letters which were not removed until the step when rewriting on the left removes something added from the right, which is achieved by replacing the word $x_j \in X_j$ with $y_j \in Y_j$. The word x_j spans over $\bar{w}^{\ell}$, w_3 and a certain nonempty prefix of $\bar{w}^{r}$; denote this prefix by $\hat{w}$, so that $x_j = \bar{w}^{\ell} w_3 \hat{w}$. Then we also have $\bar{w}^{r} = \hat{w}\tilde{w}$. Further, the replacement of $x_i \in X_i$ by $y_i \in Y_i$ and the replacement of $u_k \in U_k$ by $v_k \in V_k$ in this derivation are the rules where the letters neighbouring in w_1 with w_3 are modified. In particular, we have $\bar{w}^{\ell} \in K_i^{\ell}$ and $\bar{w}^{r} \in K_k^{r}$. When we set $q = \delta_{ij}^{\ell}(q_0, w_3)$, we immediately obtain $w_3 \in L_{ijq}^{\ell}$, which implies $\hat{w} \in (\bar{w}^{\ell} w_3)^{-1} \cdot X_j \subseteq (K_i^{\ell} \cdot L_{ijq}^{\ell})^{-1} \cdot X_j$, and therefore $\tilde{w} \in \hat{w}^{-1} \cdot K_k^{r} \subseteq ((K_i^{\ell} \cdot L_{ijq}^{\ell})^{-1} \cdot X_j)^{-1} \cdot K_k^{r}$. Since we have $w_1^{\ell} \in N_i^{\ell}$ and $w_1^{r} \in N_k^{r}$ and the word w_2 can be derived from $y_j\tilde{w}$, we can deduce that $(w_1, w_2) \in \sigma_{ijkq}^{\ell}$, which completes the proof of this case. The symmetric case of a derivation where some letter added from the left is later removed from the right can be handled dually using relations σ_{ijkq}^{r}.

Conversely, assume that $(w_1, w_2) \in \rho$. If $(w_1, w_2) \in \rho_{ik}$ for some i and k, then it is easy to construct a derivation of the form (1), which shows that $w_1 \Longrightarrow^* w_2$.

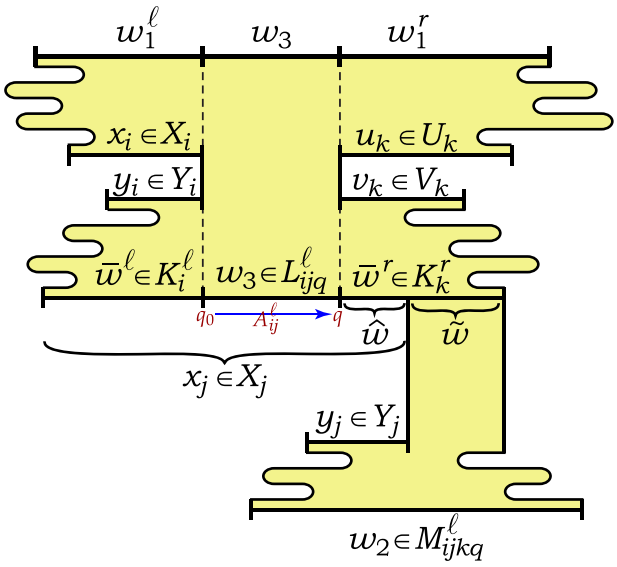

Figure 1: $(w_1^\ell w_3 w_1^r, w_2) \in \sigma_{ijkq}^\ell$ in the rationality proof.

So let us assume that $(w_1, w_2) \in \sigma_{ijkq}^\ell$ for some $i, j, k \in \{1, \ldots, n\}$ and $q \in Q_{ij}^\ell$ (the case $(w_1, w_2) \in \sigma_{ijkq}^r$ is again handled dually). Then $w_1 = w_1^\ell w_3 w_1^r$, where $w_1^\ell \in N_i^\ell$, $w_3 \in L_{ijq}^\ell$ and $w_1^r \in N_k^r$. Additionally, we have $y_j \tilde{w} \Longrightarrow^* w_2$ for certain words $y_j \in Y_j$ and $\tilde{w} \in ((K_i^\ell \cdot L_{ijq}^\ell)^{-1} \cdot X_j)^{-1} \cdot K_k^r$. Hence, there exist $\bar{w}^r \in K_k^r$ and $\hat{w} \in (K_i^\ell \cdot L_{ijq}^\ell)^{-1} \cdot X_j$ satisfying $\bar{w}^r = \hat{w}\tilde{w}$. Consequently, we have $w_4 \hat{w} \in (K_i^\ell)^{-1} \cdot X_j$ for some $w_4 \in L_{ijq}^\ell$. In particular, this means that $\delta_{ij}^\ell(q_0, w_4) = q = \delta_{ij}^\ell(q_0, w_3)$ and therefore $w_3 \hat{w} \in (K_i^\ell)^{-1} \cdot X_j$ too. This provides us with words $x_j \in X_j$ and $\bar{w}^\ell \in K_i^\ell$ satisfying $x_j = \bar{w}^\ell w_3 \hat{w}$. Taking any words $y_i \in Y_i$ and $v_k \in V_k$ such that $y_i \Longrightarrow^* \bar{w}^\ell$ and $v_k \Longrightarrow^* \bar{w}^r$, we can now derive

$$w_1 = w_1^\ell w_3 w_1^r \Longrightarrow^* x_i w_3 u_k \Longrightarrow^2 y_i w_3 v_k \Longrightarrow^* \bar{w}^\ell w_3 \bar{w}^r = \bar{w}^\ell w_3 \hat{w}\tilde{w} = x_j \tilde{w} \Longrightarrow y_j \tilde{w} \Longrightarrow^* w_2.$$

This concludes the proof. □

The value of Theorem 2.4 comes from the importance of rational relations, see e.g. [8], or [10], where the power of rational relations is utilized. Unfortunately, for the one-way rewriting of the next section we do not have a counterpart of the above rationality result.

3 One-way rewriting

In the one-way rewriting, a prefix x is consumed and a suffix y is appended at every step. If the exact set of admissible pairs (x, y) can be specified, this is rewriting of the type studied by Post [24, 25], and his results

imply its computational universality, which is explained in detail in [19]. Post also gave a simple example of one-way rewriting with the set of pairs $\{(0ab, 00), (1ab, 1101) \mid a, b \in \{0, 1\}\}$, where the behaviour remains unknown up to now.

Let us consider the *uncontrolled* version, where the choice of x is independent of the choice of y. Formally, let $I, X, Y \subseteq \Sigma^*$ be regular sets of initial words, words read from the left and words written at the right, respectively. The binary relation of *one-step derivability* on the set Σ^* is defined as follows: for every $x \in X$, $y \in Y$ and $w \in \Sigma^*$, $xw \Longrightarrow wy$. The reflexive and transitive closure of this relation is the relation of *derivability*. A word is generated if it is derived from a word in I in zero or more steps.

Intuitively, it is feasible to think that the languages obtained by one-way rewriting are not necessarily regular. Indeed, the "storage" used in the process of rewriting is a queue of an unbounded size, and its contents need to be remembered. However, in the next theorem we establish the regularity of generated languages.

Theorem 3.1. *The language generated by any uncontrolled one-way rewriting system is regular, and, given finite automata for I, X and Y, a finite automaton for this language can be effectively constructed.*

The proof is by a reduction to controlled one-sided rewriting. One-way rewriting starts with a word from I and proceeds by biting off its prefixes and appending words taken from the set Y to the right. Once the initial word is consumed, the elements of Y earlier appended to the right start appearing at the left, and a typical derivation step is as in Figure 2(a): a word u, zero or more words from Y and a prefix of another word from Y are consumed (with their concatenation being in X), and a new word from Y is appended to the end. For the subsequent construction it is important that u is always either a suffix of the initial word (this is the case at the last step of its consumption) or a suffix of some word from Y.

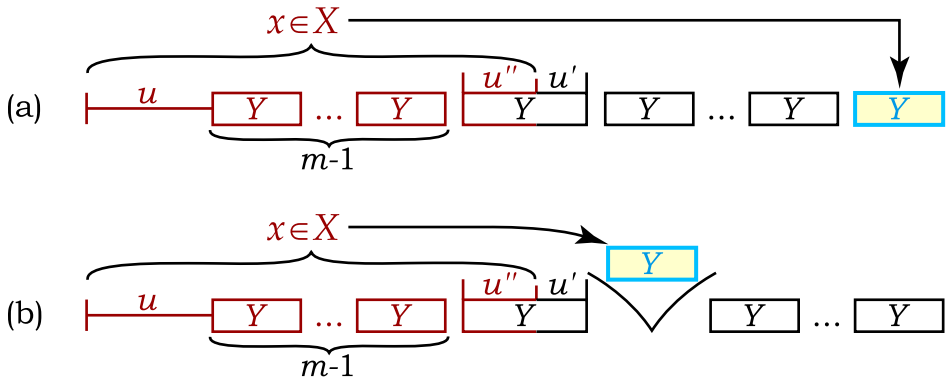

Figure 2: (a) Uncontrolled one-way rewriting and (b) its simulation from the left side.

Notice that the choice of a particular element of Y becomes relevant only when it appears at the left and is consumed, and since the choice is uncontrolled, it can be postponed till the moment of consumption. The right part of the word can be assumed to contain identical bricks labelled Y, as in the figure. Then the derivation step shown in Figure 2(a) can be equivalently reformulated by inserting a new brick near the left end rather than appending it to the right, which is shown in Figure 2(b). The latter type of rules can be simulated using one-sided rewriting.

Consider DFAs for the languages I and Y, and, in order to simplify the notation, let us merge them into a single DFA $(\Sigma, Q, q_0^I, q_0^Y, \delta, F)$ with two initial states, q_0^I and q_0^Y. For every $q \in Q$, let $L(q) = \{w \mid \delta(q, w) \in F\}$. Now $L(q_0^I) = I$ and $L(q_0^Y) = Y$. Let $\#$ be a symbol not in Q, which will be used to denote an arbitrary word from Y. Construct a one-sided controlled rewriting system over the alphabet $Q \cup \{\#\}$ with a single one-letter initial word q_0^I and with the following rules:

I. for all $q, q' \in Q$, such that $\delta(q, x) = q'$ for some $x \in X$, there is a rule $q \xrightarrow{\ell} q'\#$;

II. for all $q, q' \in Q$ and for all $m > 0$, if, for some $x_1x_2x_3 \in X$, $x_1 \in L(q)$, $x_2 \in Y^{m-1}$ and $\delta(q_0^Y, x_3) = q'$, then there is a rule $q\#^m \xrightarrow{\ell} q'\#$.

It is clear that the rules of the first type can be effectively constructed: there are at most $|Q|^2$ such rules, and for every pair (q, q') it is enough to test the nonemptiness of the intersection of two regular languages, namely X and $\{w \mid \delta(q, w) = q'\}$.

The second case, which is illustrated in Figure 2, is not as obvious, because for each pair (q, q') there may exist infinitely many suitable ms that accordingly require an infinite set of rules. Still the set of rules is recognizable, because, for every (q, q'), the set of all ms, such that $X \cap L(q) \cdot Y^{m-1} \cdot \{u \mid \delta(q_0^Y, u) = q'\} \neq \varnothing$, is ultimately periodic. This is constructively proved in the following lemma.

Lemma 3.2. *For any regular languages K, L, M, N the set S of all non-negative integers satisfying the condition $K \cap LM^nN \neq \varnothing$ is ultimately periodic and algorithmically computable.*

Proof. Let $(\Sigma, P, p_0, \rho, F_K)$ be a DFA recognizing K and let k be the number of its states. We consider a binary relation μ on P describing which states of the automaton can be connected using words from the language M. The relation μ is defined for any states $p, p' \in P$ by the rule

$$(p, p') \in \mu \iff \exists w \in M \colon \rho(p, w) = p' .$$

Then there exists a positive integer $\ell \leqslant 2^{k^2}$ such that μ^ℓ is idempotent, i.e. $\mu^{2\ell} = \mu^\ell$. Let us verify that, for every $n \geqslant \ell$, we have $n \in S$ if and only

if $n + \ell \in S$. Assuming that $n \in S$, there exist words $u \in L$, $v \in N$ and $w_1, \ldots, w_n \in M$ satisfying $uw_1 \cdots w_n v \in K$. Then

$$(\rho(p_0, u), \rho(p_0, uw_1 \cdots w_\ell)) \in \mu^\ell = \mu^{2\ell},$$

and so one can find words $\bar{w}_1, \ldots, \bar{w}_{2\ell} \in M$ such that $\rho(p_0, uw_1 \cdots w_\ell) = \rho(p_0, u\bar{w}_1 \cdots \bar{w}_{2\ell})$. Therefore,

$$\rho(p_0, u\bar{w}_1 \cdots \bar{w}_{2\ell} w_{\ell+1} \cdots w_n v) = \rho(p_0, uw_1 \cdots w_n v),$$

and thus $u\bar{w}_1 \cdots \bar{w}_{2\ell} w_{\ell+1} \cdots w_n v \in K$.

The converse implication can be verified analogously. Altogether, the set S is ultimately periodic with period ℓ. Since ℓ is bounded, the set S can be found algorithmically. □

Next we prove the main lemma showing the correctness of our simulation.

Lemma 3.3. *A word $w \in \Sigma^*$ can be derived in $t \geqslant 0$ steps in the original one-way rewriting system if and only if there exists a state $q \in Q$ and a number $k \geqslant 0$, such that $q\#^k$ can be derived in t steps in the constructed one-sided rewriting system, and $w \in L(q)Y^k$.*

Proof. Lemma is proved by induction on t, the length of both derivations. The basis, $t = 0$, is clear: the only word that can be derived in the constructed system in 0 steps is q_0^I, and $L(q_0^I)$ equals I, the set of words derivable in the original system in 0 steps. So let us move to the induction step.

Suppose the word $w' \in \Sigma^*$ is derivable in $t + 1$ steps in the original system, and let us prove that it can be represented in the form $L(q')Y^{k'}$ for some $q'\#^{k'}$ derivable in the constructed rewriting system. Since $w' \in \Sigma^*$ is derivable in $t + 1$ steps, there exists a word w derivable in t steps, such that $w \Longrightarrow w'$. By the induction hypothesis for the t-step derivation of w, there exist q and k, such that $q\#^k$ is derivable in t steps in the constructed system and $w \in L(q)Y^k$. The latter means that there exists a factorization $w = uy_1 \ldots y_k$, where $u \in L(q)$ and $y_1, \ldots, y_k \in Y$.

Consider the rewriting of w by w' at the $(t+1)$-th step, in which a prefix $x \in X$ is removed from w, and instead a word $y \in Y$ is appended to the right end. There are two cases depending on the length of the prefix x.

Case I. The word x is a prefix of u, i.e., $u = xu'$ and the derivation is of the form

$$w = xu'y_1 \ldots y_k \Longrightarrow u'y_1 \ldots y_k y = w',$$

where $y \in Y$.

Let $q' = \delta(q, x)$. The constructed rewriting system contains the rule $q \xrightarrow{\ell} q'\#$, and hence we can derive $q\#^k \Longrightarrow q'\#^{k+1}$. Since $\delta(q', u') = \delta(q, xu') \in F$, we see that $u' \in L(q')$. Therefore, the word w' is in $L(q')Y^{k+1}$, which completes the proof of this case.

Case II. The word x spans over $uy_1 \dots y_{\ell-1}$, for $\ell \geqslant 1$, and then cuts y_ℓ in two parts. Then $x = uy_1 \dots y_{\ell-1}u''$ and $y_\ell = u''u'$, and the derivation is

$$w = uy_1 \dots y_{\ell-1}u''u'y_{\ell+1} \dots y_k \Longrightarrow u'y_{\ell+1} \dots y_k y = w',$$

where $y \in Y$ is the word appended to the right.

Let $q' = \delta(q_0^Y, u'')$. The one-sided rewriting system, contains the rule $q\#^\ell \xrightarrow{\ell} q'\#$, which allows one to derive $q\#^k \Longrightarrow q'\#^{k-\ell+1}$. Since $\delta(q', u') = \delta(q_0^Y, u''u') = \delta(q_0^Y, y_\ell) \in F$, we know that $u' \in L(q')$. The word w' is therefore in $L(q')Y^{k-\ell+1}$, and the second case is proved.

In order to prove the converse implication, let $k \geqslant 0$ and $q' \in Q$, let $q'\#^k$ be derivable in the constructed one-sided rewriting system in $t+1$ steps, and let $w' \in L(q') \cdot Y^k$. It is claimed that w' is derivable in $t+1$ steps in the original one-way rewriting system.

Let us represent w' as $uy_1 \dots y_k$, where $u \in L(q')$ and $y_1, \dots, y_k \in Y$. Consider the last step in the derivation of $q'\#^k$, which, by the construction of the one-sided rewriting system, can be of one of the two types:

Case I. Suppose $q'\#^k$ is derived using a rule of the form $q \xrightarrow{\ell} q'\#$. Then it is derived from $q\#^{k-1}$, which itself is derivable in t steps. Since there is a rule $q \to q'\#$, by the construction, there must exist $x \in X$, such that $\delta(q, x) = q'$.

Consider the word $w = xuy_1 \dots y_{k-1}$, which is in $L(q) \cdot Y^{k-1}$, since $\delta(q, xu) = \delta(q', u) \in F$. Since $q\#^{k-1}$ is derivable in t steps in the constructed system, by the induction hypothesis, w is derivable in t steps in the original system. Appending one more step to this derivation, we obtain w':

$$w = xuy_1 \dots y_{k-1} \Longrightarrow uy_1 \dots y_{k-1}y_k = w'.$$

Case II. Let $q'\#^k$ be derived from $q\#^{k+\ell-1}$ using a rule $q\#^\ell \to q'\#$. By the construction, there exists $x_1x_2x_3 \in X$, such that $x_1 \in L(q)$, $x_2 \in Y^{\ell-1}$ and $\delta(q_0^Y, x_3) = q'$.

Consider the word $w = x_1x_2x_3uy_1 \dots y_{k-1}$. As in the previous case, $\delta(q_0^Y, x_3u) = \delta(q', u) \in F$, meaning $x_3u \in Y$, hence $w \in L(q)Y^{\ell-1}YY^{k-1} = L(q)Y^{k+\ell-1}$. Now the induction hypothesis can be applied for w' and $q\#^{k+\ell-1}$ (the latter is derivable in t steps by assumption) to obtain that w is derivable in the original one-way system in t steps. This derivation can be extended as follows:

$$w = x_1x_2x_3u'y_1 \dots y_{k-1} \Longrightarrow uy_1 \dots y_{k-1}y_k = w'.$$

This completes the proof. □

Now the regularity of any language generated by uncontrolled one-way rewriting systems can be inferred from the regularity of any language generated by one-sided local rewriting.

of Theorem 3.1. Given $I, X, Y \subseteq \Sigma^*$, construct the rewriting system over $Q \cup \{\#\}$ as specified above. Let $K \subseteq Q \cdot \#^*$ be the language it generates.

Define a regular substitution $f\colon (Q \cup \{\#\})^* \to 2^{\Sigma^*}$ as $f(q) = L(q)$ for all $q \in Q$ and $f(\#) = Y$, and consider the language

$$f(K) = \bigcup_{q\#^k \in K} L(q) \cdot Y^k \subseteq \Sigma^*. \tag{2}$$

By Theorem 2.1, K is regular, and therefore $f(K)$ is regular as well. According to Lemma 3.3, $f(K)$ is exactly the language generated by the original one-way rewriting system. □

It is interesting to note that, in spite of the regularity result of Theorem 3.1, the derivability relation of such a rewriting system is, in general, not rational.

Example 3.4. *Let $\Sigma = \{a, b\}$ and consider the one-way rewriting system with $X = Y = \{a\}$. Its derivability relation $R \subseteq \Sigma^* \times \Sigma^*$ is not rational.*

To see this, consider, for every $m, n \geqslant 0$, the pair $(a^m b^n, b^n a^m) \in R$. Supposing that R is rational, we can use the pumping lemma for rational relations [3, Lemma 3.3] to see that $(a^{m+k} b^n, b^{n+\ell} a^m)$ must be in R for some sufficiently large m, n and for some $k + \ell > 0$. This, however, contradicts with the definition of R.

4 Controlled one-way rewriting

The regularity result of the previous section essentially relies upon the complete independence of the choice of a word being erased at the left and of a word simultaneously appended at the right. It turns out that the least restriction on this choice is enough to generate a nonregular language, in fact, even a non-context-free one.

Example 4.1. *Let $\Sigma = \{a, b, c\}$, define $I = \{abc\}$ and $X = \{ab, baba, cb, bc\} = Y$, and impose a single restriction that whenever ab is erased, the word appended should be $baba$. The language generated by this system, intersected with $(ab)^*c$, yields $\{(ab)^{2^n} c \mid n \geqslant 0\}$.*

To establish that, let us show that the first word from $(ab)^*c$ encountered in any derivation starting from $(ab)^n c$, for $n \geqslant 1$, is the word $(ab)^{2n}c$. The first n steps of the computation are deterministic: $(ab)^n c \Longrightarrow (ab)^{n-1} cbaba \Longrightarrow \ldots \Longrightarrow c(baba)^n$. Once the word $c(baba)^n = cb(ab)^{2n-1}a$ is reached, there are four choices:

1. $cb(ab)^{2n-1}a \Longrightarrow (ab)^{2n-1}aab$. This will be followed by a deterministic computation: $(ab)^{2n-1}aab \Longrightarrow \ldots \Longrightarrow aab(baba)^{2n-1}$. At this point the computation gets stuck, because no rule is applicable;

2. $cb(ab)^{2n-1}a \Longrightarrow (ab)^{2n-1}ababa = (ab)^{2n+1}a$. This results in an infinite deterministic computation $(ab)^{2n+1}a \Longrightarrow (ab)^{2n}ababa = (ab)^{2n+2}a$, $(ab)^{2n+2}a \Longrightarrow (ab)^{2n+3}a$, etc.;

3. $cb(ab)^{2n-1}a \Longrightarrow (ab)^{2n-1}acb$. Like in case 1, the computation gets stuck regardless of what rule is used;

4. $cb(ab)^{2n-1}a \Longrightarrow (ab)^{2n-1}abc = (ab)^{2n}c$, which is the intended route of the computation.

If we allow X and Y to be different, then the rewriting system with rules $X = \{ab, cb\}$, $Y = \{baba, bc\}$ and the same $ab \to baba$ restriction forms a smaller example with the above behaviour.

5 Two-way rewriting

In two-way rewriting, we are given regular or even finite sets of words $I, X \subseteq \Sigma^*$. At every step, some word belonging to X is removed either from the beginning or from the end of the word, and some word from X is appended to the other side. Formally, the binary relation of *one-step derivability* on the set Σ^* is defined as follows: for every $x, y \in X$ and $w \in \Sigma^*$, $xw \Longrightarrow wy$ and $wx \Longrightarrow yw$.

The languages generated by these rewriting systems appear to be much more complicated than in the one-way case. It is even not known whether they are in general recursive. We show that these systems can generate non-regular languages. The argument is based on the fact that these systems are powerful enough to significantly modify positions of letters in the word even when all the rules preserve the difference between the number of occurrences of two letters.

Example 5.1. *Let $\Sigma = \{a, b\}$ and let $X = \{a, aab\}$. Then the set L of all words derivable from the word ab using the set of rewriting rules*

$$\{xw \Longrightarrow wy, wx \Longrightarrow yw \mid x, y \in X,\ w \in \{a, b\}^*\}$$

is not linear context-free.

Let us consider the non-linear context-free language $P = \{a^m b^{m+n} a^n \mid m, n \geq 0, m+n \geq 1\}$. We are going to verify that $L \cap a^*b^*a^* = P$, which will show that L is not linear context-free, since linear context-free languages are closed under taking intersections with regular languages.

First, notice that L consists only of words where the number of occurrences of a is the same as that of b, because this property is preserved by all rewriting rules. This, in particular, means that $L \cap a^*b^*a^* \subseteq P$. On the

other hand, every word $b^n a^n$, for $n \geqslant 1$, can be inductively derived from ab as follows:

$$b^n a^{n-1}\underline{a} \Longrightarrow \underline{a}ab \cdot b^n a^{n-1} \Longrightarrow \underline{a}bb^n a^{n-1} \cdot a \Longrightarrow bb^n a^{n-1}a \cdot a = b^{n+1}a^{n+1}\,.$$

Since the rule $wa \Longrightarrow aw$ allows us to shift in any word $b^n a^n$ arbitrarily many as to the left, we can obtain from ab all words belonging to P. Therefore $L \cap a^*b^*a^* = P$.

6 Conclusions and open problems

We have studied a few basic modes of word rewriting. Our main result is that the language generated by any uncontrolled one-way rewriting system is regular. We also made a certain contribution to the study of local rewriting. The main question left open is the exact power of uncontrolled two-way rewriting.

As we hinted in the introduction, each of these rewriting systems also has a natural counterpart among systems of language inequalities with concatenation as the only operation, in which constants are finite or regular languages. In view of the recently grown interest in language equations in a broader sense [1, 20, 21] and their computational properties, these parallels between equations and rewriting provided an additional motivation for our study. Although no formal connection has been found, their similarity amazingly matches their expressive power:

- Local rewriting at one end resembles inequalities $XZ \subseteq YZ$, where Z is a variable. Largest solutions of these inequalities are known to be regular [1, 26]. Our results on the regularity of languages generated by local rewriting at both ends suggest to study systems of the form $\{XZ \subseteq YZ,\ ZX \subseteq ZY\}$: there seem to be no results on the regularity of their solutions so far.

- Largest solutions of inequalities of the form $XZ \subseteq ZY$, corresponding to one-way rewriting studied in Section 3, are also regular [16]. Similarly to the negative result of Section 4 on controlled one-way rewriting, largest solutions of systems $\{XZ \subseteq ZY,\ X'Z \subseteq ZY'\}$ can be, in general, non-recursively enumerable [18].

- Finally, the two-way rewriting system of Section 5 is an analogue of Conway's commutation equation $XZ = ZX$, where the largest solution is, even for a certain finite language X, also non-recursively enumerable, see [17].

We conclude by noting that there exist some natural intermediate cases between one-way and two-way rewritings. Consider rewriting systems with

rules $xw \Longrightarrow yw$ and $xw \Longrightarrow wy$ for all $x, y \in X$ (let us call it the *one-and-half-way rewriting of sending*), or with rules $xw \Longrightarrow yw$ and $wx \Longrightarrow yw$ for all $x, y \in X$ (*one-and-half-way rewriting of receiving*). While Example 5.1 can be modified to show the nonregularity in the former case, nothing seems to be known about the latter case. These types of rewriting are suggested for further study.

Acknowledgements

We are grateful to an anonymous referee, who suggested Example 3.4 and proposed the question of rationality of the relations of Section 2, which we solved in Theorem 2.4. We also wish to thank another referee for pointing us to the existing results related to two-sided rewriting.

Research supported by the Academy of Finland under grants 206039 and 208414. A preliminary version of this paper was presented at Workshop on Semigroups and Automata held in Lisbon, Portugal, July 16, 2005.

References

[1] F. Baader, R. Küsters, "Unification in a description logic with transitive closure of roles", *Logic for Programming, Artificial Intelligence, and Reasoning* (LPAR 2001, Havana, Cuba, December 3–7, 2001), LNCS 2250, 217–232.

[2] M. Benois, Parties rationnelles du groupe libre, *C. R. Acad. Sci. Paris Series A*, 269 (1969) 1188–1190.

[3] J. Berstel, *Transductions and Context-Free Languages*, BG Teubner, Stuttgart, 1979.

[4] R. V. Book, F. Otto, *String-Rewriting Systems*, Springer, 1993.

[5] J. R. Büchi, "Regular canonical systems", *Arch. Math. Logic Grundlagenforsch*, 6 (1964), 91–111.

[6] J. R. Büchi, W. H. Hosken, "Canonical systems which produce periodic sets", *Mathematical Systems Theory*, 4:1 (1970), 81–90.

[7] D. Caucal, On the regular structure of prefix rewriting, *Theoretical Computer Science*, 106 (1992), 61–86.

[8] Ch. Choffrut, *Contribution à l'étude de quelques familles remarquables de fonctions rationnelles*, Thèse d'Etat, Université Paris VII, 1978.

[9] J. H. Conway, *Regular Algebra and Finite Machines*, Chapman and Hall, 1971.

[10] Ch. Choffrut, T. Harju, J. Karhumäki, "A note on decidability questions of word semigroups", *Theoretical Computer Science*, 183 (1997), 83–92.

[11] M. A. Harrison, *Introduction to Formal Language Theory*, Addison-Wesley, 1978.

[12] D. Hofbauer, J. Waldmann, "Deleting string rewriting systems preserve regularity", *Theoretical Computer Science*, 327:3 (2004), 301–317.

[13] J. Karhumäki, I. Petre, "Two problems on commutation of languages", in *Current Trends in Theoretical Computer Science: The Challenge of the New Century, vol. 2*, World Scientific, 2004, 477–494.

[14] J. Karhumäki, "Finite sets of words and computing", *MCU 2004*, LNCS 3354, 36–49.

[15] M. I. Kratko, "On a certain class of Post calculi", *Soviet Math. Doklady*, 6 (1965), 1544–1545.

[16] M. Kunc, "Regular solutions of language inequalities and well quasi-orders", *Theoretical Computer Science*, 348:2–3 (2005), 277–293.

[17] M. Kunc, "The power of commuting with finite sets of words", *22nd Annual Symposium on Theoretical Aspects of Computer Science* (STACS 2005, Stuttgart, Germany, February 24–26, 2005) LNCS 3404, 569–580.

[18] M. Kunc, "On language inequalities $XK \subseteq LX$", *Developments in Language Theory* (DLT 2005, Palermo, Italy, July 4–8, 2005), LNCS 3572, 327–337.

[19] M. L. Minsky, *Computation: Finite and Infinite Machines*, Prentice-Hall, 1967.

[20] A. Okhotin, "Decision problems for language equations with Boolean operations", *Automata, Languages and Programming* (ICALP 2003, Eindhoven, The Netherlands, June 30–July 4, 2003), LNCS 2719, 239–251.

[21] A. Okhotin, "On the equivalence of linear conjunctive grammars to trellis automata", *RAIRO Informatique Théorique et Applications*, 38:1 (2004), 69–88.

[22] Gh. Păun, G. Rozenberg, A. Salomaa, *DNA Computing*, Springer, 1999.

[23] J.-E. Pin, J. Sakarovitch, "Une application de la représentation matricielle des transductions", *Theoretical Computer Science*, 35 (1985), 271–293.

[24] E. L. Post, "On a simple class of deductive systems", *Bulletin of the American Mathematical Society*, 27 (1921), 396–397.

[25] E. L. Post, "Formal reductions of the general combinatorial decision problem", *American Journal of Mathematics*, 65:2 (1943), 197–215.

[26] M. O. Rabin, "Decidability of second-order theories and automata on infinite trees", *Transactions of the American Mathematical Society*, 141 (1969), 1–35.

[27] J. Sakarovitch, *Elements de théorie des automates*, Vuibert, 2003.

[28] A. Salomaa, *Formal Languages*, Academic Press, 1973.

Turku
Centre *for*
Computer
Science

Lemminkäisenkatu 14 A, 20520 Turku, Finland | www.tucs.fi

University of Turku
- Department of Information Technology
- Department of Mathematical Sciences

Åbo Akademi University
- Department of Computer Science
- Institute for Advanced Management Systems Research

Turku School of Economics and Business Administration
- Institute of Information Systems Sciences

ISBN 952-12-1658-1
ISSN 1239-1891