

# The Church-Rosser Languages Are the Deterministic Variants of the Growing Context-Sensitive Languages

Gundula Niemann and Friedrich Otto

Fachbereich Mathematik/Informatik, Universität Kassel, D-34109 Kassel  
e-mail: <niemann,otto>@theory.informatik.uni-kassel.de

**Abstract.** The growing context-sensitive languages have been classified through the shrinking two-pushdown automaton, the deterministic version of which characterizes the class of generalized Church-Rosser languages (Buntrock and Otto 1995). Exploiting this characterization we prove that this latter class coincides with the class of Church-Rosser languages that was introduced by McNaughton, Narendran, and Otto (1988). Based on this result several open problems of McNaughton et al can be answered.

## 1 Introduction

If  $R$  is a finite and length-reducing string-rewriting system on some finite alphabet  $\Sigma$ , then there exists a linear-time algorithm that, given a string  $w \in \Sigma^*$  as input, computes an irreducible descendant  $w_0$  of  $w$  with respect to the reduction relation  $\rightarrow_R^*$  that is induced by  $R$  [2, 3]. If, in addition, the system  $R$  is *confluent*, then the irreducible descendant  $w_0$  is uniquely determined by  $w$ . Hence, in this situation two strings  $u$  and  $v$  are congruent modulo the Thue congruence  $\leftrightarrow_R^*$  induced by  $R$  if and only if their respective irreducible descendants  $u_0$  and  $v_0$  coincide. Thus, the word problem for a finite, length-reducing, and confluent string-rewriting system is decidable in linear time.

Motivated by this result McNaughton, Narendran, and Otto [11] introduced the notion of a *Church-Rosser language*. A Church-Rosser language  $L \subseteq \Sigma^*$  is given through a finite, length-reducing, and confluent string-rewriting system  $R$  on some alphabet  $\Gamma$  properly containing  $\Sigma$ , two irreducible strings  $t_1, t_2 \in (\Gamma \setminus \Sigma)^*$ , and an irreducible letter  $Y \in \Gamma \setminus \Sigma$  satisfying the following condition for all strings  $w \in \Sigma^*$ :  $w \in L$  if and only if  $t_1 w t_2 \rightarrow_R^* Y$ . Hence, the membership problem for a Church-Rosser language is decidable in linear time, and so the class CRL of Church-Rosser languages is contained in the class CSL of context-sensitive languages.

On the other hand, the class CRL contains the class DCFL of deterministic context-free languages, and it contains some languages that are not even context-free [11]. Hence, the class CRL can be seen as an extension of the class DCFL that preserves the linear-time decidability of the membership problem. As such it is certainly an interesting language class.

Accordingly, McNaughton et al established some closure properties for the class CRL, but it remained open whether the class CRL is closed under the operation of complementation. Accordingly, they introduced the class of *Church-Rosser decidable languages* CRDL, which still contains the class DCFL and which is closed under complementation. Also it remained open at the time whether or not every context-free language is a Church-Rosser language, although it was conjectured that the linear language  $L_0 := \{ww^\sim | w \in \{a, b\}^*\}$  is not a Church-Rosser language. Here  $w^\sim$  denotes the reversal of the string  $w$ .

After their introduction the Church-Rosser languages did not receive much attention until another, seemingly unrelated development had taken place. Dahlhaus and Warmuth [8] considered the class GCSL of *growing context-sensitive languages*. These languages are generated by context-sensitive grammars each production rule of which is strictly length-increasing. They proved that these languages have membership problems that are decidable in polynomial time. Although it might appear from the definition that GCSL is not an interesting class of languages, Buntrock and Lorys showed that GCSL is an *abstract family of languages* [5], that is, this class of languages is closed under union, concatenation, iteration, intersection with regular languages,  $\varepsilon$ -free homomorphisms, and inverse homomorphisms. Exploiting these closure properties Buntrock and Lorys characterized the class GCSL through various other classes of grammars that are less restricted [5, 6].

Using these grammars Buntrock and Otto [7] obtained a characterization of the class GCSL by a nondeterministic machine model, the so-called *shrinking pushdown automaton with two pushdown stores* (sTPDA). The input for such a machine is provided as the initial contents of one of the pushdown stores, and it accepts either by final state or (equivalently) by empty pushdown stores. A positive weight is assigned to each tape symbol and each internal state symbol of the machine. By adding up the weights this gives a weight for each configuration. Now it is required that the weight of the actual configuration decreases with each step of the machine. It is with respect to these weights that the two-pushdown automaton is called *shrinking*.

Since the sTPDA is a nondeterministic device, it was only natural to consider the class of languages that are accepted by the deterministic variant of it. As it turned out the deterministic sTPDA accept exactly the so-called *generalized Church-Rosser languages*, which are obtained from the Church-Rosser languages by admitting finite, *weight-reducing*, and confluent string-rewriting systems in the definition [7]. Thus, the class GCRL of generalized Church-Rosser languages coincides with the class of ‘deterministic growing context-sensitive languages.’ In particular, it follows that this class is closed under complementation. Further, Buntrock and Otto concluded from this result that the language classes CFL and GCRL, and therewith the classes CFL and CRL, are indeed incomparable under set inclusion. Since CFL is contained in GCSL, it follows that GCRL is properly contained in the class GCSL, that is, we obtain the following chain of (proper) inclusions:

$$\text{DCFL} \subset \text{CRDL} \subseteq \text{CRL} \subseteq \text{GCRL} \subset \text{GCSL} \subset \text{CSL},$$

where it was left open whether or not the two inclusions  $\text{CRDL} \subseteq \text{CRL} \subseteq \text{GCRL}$  are proper.

Here we show that the three language classes  $\text{CRDL}$ ,  $\text{CRL}$ , and  $\text{GCRL}$  coincide. Our proof makes use of the above-mentioned characterization of the generalized Church-Rosser languages through the deterministic  $\text{sTPDA}$ . We will prove that each language that is accepted by some deterministic  $\text{sTPDA}$  is actually a Church-Rosser decidable language. Hence,  $\text{GCRL} \subseteq \text{CRDL}$  implying that the three classes above actually coincide. Hence, the class of Church-Rosser languages can be characterized as the class of deterministic growing context-sensitive languages.

It remains to determine the closure properties of this class of languages. The closure under the operation of taking the complement follows from the above characterization. Recently, Otto, Katsura, and Kobayashi [12] proved that the class of Church-Rosser languages is a basis for the recursively enumerable (r.e.) languages. Here, a class of languages  $\mathbb{C}$  is called a *basis* for the r.e. languages, if, for each r.e. language  $L \subseteq \Sigma^*$ , there exists a language  $C \in \mathbb{C}$  on some alphabet  $\Gamma$  strictly containing  $\Sigma$  such that  $L = \pi_\Sigma(C)$ , where  $\pi_\Sigma$  denotes the canonical projection from  $\Gamma^*$  onto  $\Sigma^*$ . It follows that the class  $\text{CRL}$  is not closed under morphisms.

This paper is organized as follows. In Section 2 we introduce the necessary notation regarding string-rewriting systems and restate the definitions of the various classes of Church-Rosser languages. In the next section we introduce the shrinking two-pushdown automaton and restate some results from Buntrock and Otto [7]. In addition we prove a technical result for this type of automaton. Then in Section 4 we prove the announced main result, and in the next section we summarize the known closure and non-closure properties of the class  $\text{CRL}$ . In the final section we review our results and draw some easy consequences.

## 2 The Church-Rosser Languages

Here we restate the main definitions and establish notation regarding the various classes of Church-Rosser languages. For additional information concerning the notions introduced the reader is asked to consult the literature, where [3] serves as our main reference concerning the theory of string-rewriting systems, and [10] is our main reference for formal language and automata theory.

Let  $\Sigma$  be a finite alphabet. Then  $\Sigma^*$  denotes the set of strings over  $\Sigma$  including the empty string  $\varepsilon$ , and  $\Sigma^+ := \Sigma^* \setminus \{\varepsilon\}$ . A function  $\varphi : \Sigma \rightarrow \mathbb{N}_+$  is called a *weight-function*. Its extension to  $\Sigma^*$ , which we will also denote by  $\varphi$ , is defined inductively through  $\varphi(\varepsilon) := 0$  and  $\varphi(wa) := \varphi(w) + \varphi(a)$  for all  $w \in \Sigma^*$  and  $a \in \Sigma$ . A particular weight-function is the *length-function*  $|\cdot| : \Sigma \rightarrow \mathbb{N}_+$ , which assigns each letter the weight (length) 1.

A *string-rewriting system*  $R$  on  $\Sigma$  is a subset of  $\Sigma^* \times \Sigma^*$ . An element  $(\ell, r) \in R$  is called a *rewrite rule* or simply a *rule*, and it will usually be written as  $(\ell \rightarrow r)$ . A string-rewriting system  $R$  induces several binary relations on  $\Sigma^*$ , the simplest

of which is the *single-step reduction relation*

$$\rightarrow_R := \{(ulv, urv) \mid u, v \in \Sigma^*, (\ell \rightarrow r) \in R\}.$$

Its reflexive and transitive closure is the *reduction relation*  $\rightarrow_R^*$  induced by  $R$ , and its reflexive, symmetric, and transitive closure  $\leftrightarrow_R^*$  is the *Thue congruence* generated by  $R$ .

If  $u \rightarrow_R^* v$ , then  $u$  is an *ancestor* of  $v$ , and  $v$  is a *descendant* of  $u$ . If there is no  $v \in \Sigma^*$  such that  $u \rightarrow_R v$  holds, then the string  $u$  is called *irreducible* (mod  $R$ ). By  $\text{IRR}(R)$  we denote the set of all irreducible strings. If  $R$  is finite, then  $\text{IRR}(R)$  is obviously a regular language.

The string-rewriting system  $R$  is called

- *length-reducing* if  $|\ell| > |r|$  holds for each rule  $(\ell \rightarrow r) \in R$ ,
- *weight-reducing* if there exists a weight-function  $\varphi$  such that  $\varphi(\ell) > \varphi(r)$  holds for each rule  $(\ell \rightarrow r) \in R$ ,
- *confluent* if, for all  $u, v, w \in \Sigma^*$ ,  $u \rightarrow_R^* v$  and  $u \rightarrow_R^* w$  imply that  $v$  and  $w$  have a common descendant.

If a string-rewriting system  $R$  is weight-reducing, then it allows no infinite reduction sequence of the form  $w_0 \rightarrow_R w_1 \rightarrow_R \dots$ ; indeed, if  $w_0 \rightarrow_R w_1 \rightarrow_R \dots \rightarrow_R w_m$ , then  $m \leq \varphi(w_0)$ . If, in addition,  $R$  is confluent, then each string  $w \in \Sigma^*$  has a unique irreducible descendant  $w_0 \in \text{IRR}(R)$ . Actually, in this situation  $u \leftrightarrow_R^* v$  if and only if  $u_0 = v_0$ . Since  $u_0$  can be determined from  $u$  in linear time, this shows that the Thue congruence  $\leftrightarrow_R^*$  is decidable in linear time for each finite, weight-reducing, and confluent string-rewriting system.

### Definition 1.

- (a) A language  $L \subseteq \Sigma^*$  is a Church-Rosser language (CRL) if there exist an alphabet  $\Gamma \supsetneq \Sigma$ , a finite, length-reducing, confluent string-rewriting system  $R$  on  $\Gamma$ , two strings  $t_1, t_2 \in (\Gamma \setminus \Sigma)^* \cap \text{IRR}(R)$ , and a letter  $Y \in (\Gamma \setminus \Sigma) \cap \text{IRR}(R)$  such that, for all  $w \in \Sigma^*$ ,  $t_1 w t_2 \rightarrow_R^* Y$  if and only if  $w \in L$ .
- (b) A language  $L \subseteq \Sigma^*$  is a Church-Rosser decidable language (CRDL) if it is a Church-Rosser language, and there exists a letter  $N \in (\Gamma \setminus \Sigma) \cap \text{IRR}(R)$  such that, for all  $w \in \Sigma^*$ ,  $t_1 w t_2 \rightarrow_R^* N$  if and only if  $w \notin L$ .
- (c) A language  $L \subseteq \Sigma^*$  is a generalized Church-Rosser language (GCRL) if there exist an alphabet  $\Gamma \supsetneq \Sigma$ , a finite, weight-reducing, confluent string-rewriting system  $R$  on  $\Gamma$ , two strings  $t_1, t_2 \in (\Gamma \setminus \Sigma)^* \cap \text{IRR}(R)$  and a letter  $Y \in (\Gamma \setminus \Sigma) \cap \text{IRR}(R)$  such that, for all  $w \in \Sigma^*$ ,  $t_1 w t_2 \rightarrow_R^* Y$  if and only if  $w \in L$ .

Analogously to (b) the class of *generalized Church-Rosser decidable languages* could be defined, but the results of Buntrock and Otto [7] imply that this class coincides with the class GCRL of generalized Church-Rosser languages.

### 3 Shrinking Two-Pushdown Automata

In [7] Buntrock and Otto introduce the following type of automaton in order to characterize the class GCSL of growing context-sensitive languages.

**Definition 2.**

- (a) A two-pushdown automaton (TPDA) is a nondeterministic automaton with two pushdown stores. It is defined as a 7-tuple  $M = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$ , where
- $Q$  is the finite set of states,
  - $\Sigma$  is the finite input alphabet,
  - $\Gamma$  is the finite tape alphabet with  $\Gamma \supsetneq \Sigma$  and  $\Gamma \cap Q = \emptyset$ ,
  - $q_0 \in Q$  is the initial state,
  - $\perp \in \Gamma \setminus \Sigma$  is the bottom marker of pushdown stores,
  - $F \subseteq Q$  is the set of final (or accepting) states, and
  - $\delta : Q \times \Gamma \times \Gamma \rightarrow 2^{Q \times \Gamma^* \times \Gamma^*}$  is the transition relation, where  $\delta(q, a, b)$  is a finite set for each triple  $(q, a, b) \in Q \times \Gamma \times \Gamma$ .
- $M$  is a deterministic two-pushdown automaton (DTPDA), if  $\delta$  is a (partial) function from  $Q \times \Gamma \times \Gamma$  into  $Q \times \Gamma^* \times \Gamma^*$ .
- (b) A configuration of a (D)TPDA  $M$  is described as  $uqv$  with  $q \in Q$  and  $u, v \in \Gamma^*$ , where  $u$  is the contents of the first pushdown store with the first letter of  $u$  at the bottom and the last letter of  $u$  at the top,  $q$  is the current state, and  $v$  is the contents of the second pushdown store with the last letter of  $v$  at the bottom and the first letter of  $v$  at the top.  $M$  induces a computation relation  $\vdash_M^*$  on the set of configurations, which is the reflexive, transitive closure of the single-step computation relation  $\vdash_M$  (see, e.g., [10]). For an input string  $w \in \Sigma^*$ , the corresponding initial configuration is  $\perp q_0 w \perp$ .  $M$  accepts by empty pushdown stores:

$$N(M) := \{w \in \Sigma^* \mid \exists q \in Q : \perp q_0 w \perp \vdash_M^* q\}.$$

- (c) A (D)TPDA  $M$  is called shrinking if there exists a weight function  $\varphi : Q \cup \Gamma \rightarrow \mathbb{N}_+$  such that, for all  $q \in Q$  and  $a, b \in \Gamma$ , if  $(p, u, v) \in \delta(q, a, b)$ , then  $\varphi(upv) < \varphi(aqb)$ . By sTPDA and sDTPDA we denote the corresponding classes of shrinking automata.

Thus, if  $M$  is a shrinking TPDA with weight-function  $\varphi$ , then  $\varphi(u_1 q_1 v_1) > \varphi(u_2 q_2 v_2)$  holds for all configurations  $u_1 q_1 v_1$  and  $u_2 q_2 v_2$  of  $M$  that satisfy  $u_1 q_1 v_1 \vdash_M u_2 q_2 v_2$ . Observe that the input is provided to a TPDA as the initial contents of its second pushdown store, and that in order to accept a TPDA is required to empty its pushdown stores. Thus, it is forced to consume the input completely. Using standard techniques from automata theory it can be shown that, for a (shrinking) (D)TPDA  $M = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$ , we may require that the special symbol  $\perp$  can only occur at the bottom of a pushdown store, and that no other symbol can occur at that place.

From the definition of the transition relation  $\delta$  we see that  $M$  halts immediately whenever one of its pushdown stores is emptied. Because of the above property this happens if and only if a transition of the form  $(q, a, \perp) \mapsto (q', \alpha, \varepsilon)$  or  $(q, \perp, b) \mapsto (q', \varepsilon, \beta)$  is performed. Thus, we can assume without loss of generality that, if  $M$  does accept on input  $w \in \Sigma^*$ , then  $\perp q_0 w \perp \vdash_M^* q$  for some  $q \in F$ , and if  $M$  does not accept on input  $w \in \Sigma^*$ , then  $\perp q_0 w \perp \vdash_M^* \perp q$  for some  $q \in F$ , that is, even in this situation  $M$  empties its second pushdown store completely and only leaves the bottom marker on its first pushdown store before it halts. Hence, all the halting and accepting configurations of  $M$  are of the form  $q$ , where  $q \in F$ , and all the halting and rejecting configurations of  $M$  are of the form  $\perp q$ , where  $q \in F$ . In addition, we can assume that  $M$  only has a single halting state.

Buntrock and Otto established the following characterization for the classes of languages that are accepted by nondeterministic or deterministic shrinking TPDA's, respectively.

**Proposition 3.** [7]

- (a) *A language is accepted by some shrinking TPDA if and only if it is growing context-sensitive.*
- (b) *A language is accepted by some shrinking DTPDA if and only if it is a generalized Church-Rosser language.*

A detailed presentation of the class GCSL of growing context-sensitive languages can be found in Buntrock's Habilitationsschrift [4]. The above proposition shows that the generalized Church-Rosser languages can be interpreted as the deterministic variants of the growing context-sensitive languages.

We close this section with a technical lemma on shrinking TPDA that we will need in the next section to prove our main result.

**Lemma 4.** *Let  $M$  be a TPDA that is shrinking with respect to the weight-function  $\varphi$ . Then there exists a TPDA  $M'$  accepting the same language as  $M$  such that  $M'$  is deterministic, if  $M$  is, and  $M'$  is shrinking with respect to a weight-function  $\psi$  that satisfies the following condition:*

- (\*) *Whenever  $u_1 q_1 v_1$  and  $u_2 q_2 v_2$  are configurations of  $M'$  such that  $u_1 q_1 v_1 \vdash_{M'}$   $u_2 q_2 v_2$ , then  $\psi(u_1 q_1 v_1) - \psi(u_2 q_2 v_2) = 1$ .*

*Proof.* Let  $M = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$  be a TPDA that is shrinking with respect to the weight-function  $\varphi : Q \cup \Gamma \rightarrow \mathbb{N}_+$ , that is,  $\varphi(aqb) - \varphi(upv) > 0$  for all  $q \in Q$ ,  $a, b \in \Gamma$ , and  $(p, u, v) \in \delta(q, a, b)$ . We construct a TPDA  $M' := (Q', \Sigma, \Gamma, \delta', q_0, \perp, F)$  and a weight-function  $\psi : Q' \cup \Gamma \rightarrow \mathbb{N}_+$  as follows.

First we number the instructions of  $M$ , that is, the lines in the table describing the transition relation  $\delta$ , from 1 to  $m$ . For each  $i \in \{1, \dots, m\}$ , let the  $i$ -th instruction of  $M$  be denoted as  $(p_i, u_i, v_i) \in \delta(q_i, a_i, b_i)$ , and let  $\gamma_i := \varphi(a_i q_i b_i) - \varphi(u_i p_i v_i)$ .

If  $\gamma_i = 1$ , then take  $Q'_i := \emptyset$  and add the transition  $(q_i, a_i, b_i) \rightarrow (p_i, u_i, v_i)$  to  $\delta'$ . If  $\gamma_i > 1$ , then take  $Q'_i := \{q_{i,1}, \dots, q_{i,\gamma_i-1}\}$ , where  $q_{i,1}, \dots, q_{i,\gamma_i-1}$  are  $\gamma_i - 1$  new states, and add the following transitions to  $\delta'$ :

$$\begin{aligned} (q_i, a_i, b_i) &\rightarrow (q_{i,1}, a_i, b_i), \\ (q_{i,j}, a_i, b_i) &\rightarrow (q_{i,j+1}, a_i, b_i), \quad j = 1, \dots, \gamma_i - 2, \\ (q_{i,\gamma_i-1}, a_i, b_i) &\rightarrow (p_i, u_i, v_i). \end{aligned}$$

Finally, let  $Q' := Q \cup \bigcup_{i=1}^m Q'_i$ , let  $\delta'$  consist of all the transitions introduced so far, and define a preliminary weight-function  $\psi' : Q' \cup \Gamma \rightarrow \mathbb{Z}$  as follows:

$$\begin{aligned} \psi'(a) &:= \varphi(a) && \text{for all } a \in \Gamma, \\ \psi'(q_i) &:= \varphi(q_i) && \text{for all } q_i \in Q, \\ \psi'(q_{i,j}) &:= \varphi(q_i) - j && \text{for all } i \in \{1, \dots, m\} \text{ and } j \in \{1, \dots, \gamma_i - 1\}. \end{aligned}$$

It is easily verified that  $\psi'(u_1 q_1 v_1) - \psi'(u_2 q_2 v_2) = 1$  holds for all configurations  $u_1 q_1 v_1$  and  $u_2 q_2 v_2$  of  $M'$  that satisfy  $u_1 q_1 v_1 \vdash_{M'} u_2 q_2 v_2$ . Unfortunately,  $\psi'$  may not be an acceptable weight-function, since  $\psi'(q_{i,j})$  could be a negative number for some choices of  $i$  and  $j$ .

To correct this problem let  $\mu := \min\{\psi'(p') \mid p' \in Q'\}$ . If  $\mu < 0$ , then choose  $\psi(q') := \psi'(q') + |\mu| + 1$  for all  $q' \in Q'$ , otherwise, let  $\psi(q') := \psi'(q')$  for all  $q' \in Q'$ . Also choose  $\psi(a) := \psi'(a)$  for all  $a \in \Gamma$ . Then  $\psi : Q' \cup \Gamma \rightarrow \mathbb{N}_+$  is a weight-function such that  $\psi(u_1 q_1 v_1) - \psi(u_2 q_2 v_2) = 1$  holds for all configurations  $u_1 q_1 v_1$  and  $u_2 q_2 v_2$  of  $M'$  that satisfy  $u_1 q_1 v_1 \vdash_{M'} u_2 q_2 v_2$ .

It is easily seen that  $N(M') = N(M)$  and that  $M'$  is deterministic, if  $M$  is deterministic.  $\square$

Thus, in the following we can always assume that in each step of a sTPDA the weight of the actual configuration decreases by 1. Hence, if  $u_1 q_1 v_1$  and  $u_2 q_2 v_2$  are configurations of an sTPDA  $M$  with weight-function  $\varphi$  such that  $u_1 q_1 v_1 \vdash_M^k u_2 q_2 v_2$  for some  $k \in \mathbb{N}$ , then  $\varphi(u_1 q_1 v_1) - \varphi(u_2 q_2 v_2) = k$ .

## 4 The Main Result

From the definitions we know that  $\text{CRDL} \subseteq \text{CRL} \subseteq \text{GCRL}$  holds. Here we prove that also  $\text{GCRL} \subseteq \text{CRDL}$  holds, thus showing that the three classes actually coincide.

**Theorem 5.**  $\text{GCRL} \subseteq \text{CRDL}$ .

*Proof.* Let  $L \subseteq \Sigma^*$  be a generalized Church-Rosser language. By Proposition 3(b) there exist a DTPDA  $M = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$  and a weight-function  $\varphi$  such that  $N(M) = L$ , where  $M$  is shrinking with respect to  $\varphi$ . As observed in the previous section we can assume the following:

- (i) Each non-halting configuration of  $M$  is of the form  $\perp uqv \perp$  for some  $u, v \in (\Gamma \setminus \{\perp\})^*$  and  $q \in (Q \setminus F)$ .
- (ii)  $F = \{q_f\}$ , that is,  $M$  has a single halting state only.
- (iii) The only accepting and halting configuration of  $M$  that is reachable from an initial configuration is the configuration  $q_f$ .
- (iv) The only non-accepting and halting configuration of  $M$  that is reachable from an initial configuration is the configuration  $\perp q_f$ .
- (v) If  $u_1 q_1 v_1 \vdash_M u_2 q_2 v_2$ , then  $\varphi(u_1 q_1 v_1) - \varphi(u_2 q_2 v_2) = 1$  (Lemma 4).

Let  $\#$  be a new symbol. We define a morphism  $h : (\Gamma \cup Q)^* \rightarrow (\Gamma \cup Q \cup \{\#\})^*$  by taking  $h(a) := a\#\varphi(a)^{-1}$  for all  $a \in \Gamma \cup Q$ . Then  $|h(w)| = \varphi(w)$  for all  $w \in (\Gamma \cup Q)^*$ , and  $h(\Gamma \cup Q) \subseteq (\Gamma \cup Q \cup \{\#\})^+$  is a prefix code. Thus, the morphism  $h : (\Gamma \cup Q)^* \rightarrow (\Gamma \cup Q \cup \{\#\})^*$  is an injective mapping. Further, let  $\mu := \max\{\varphi(a) \mid a \in \Gamma \cup Q\}$  denote the maximal weight of any letter from  $\Gamma \cup Q$ .

In order to show that the language  $L$  is actually Church-Rosser decidable, we now construct a finite, length-reducing, and confluent string-rewriting system  $R$  on some finite alphabet  $\Delta \supseteq \Sigma$  that will witness this fact. Essentially  $R$  will simulate the computations of the sDTPDA  $M$ . However, this cannot be a straightforward simulation, since  $R$  is length-reducing, while  $M$  is shrinking only with respect to the weight-function  $\varphi$ . Therefore we would like to replace a configuration  $\perp uqv \perp$  of  $M$  by the string  $h(\perp uqv \perp)$ . Since this replacement increases the length of the string considered, we need to compress the resulting string by combining several letters into a single new letter. This, however, creates another problem. If  $\perp u_1 q_1 v_1 \perp \vdash_M \perp u_2 q_2 v_2 \perp$ , then by (v)  $|h(\perp u_1 q_1 v_1 \perp)| - 1 = |h(\perp u_2 q_2 v_2 \perp)|$ , but for the compressed forms of the strings  $h(\perp u_1 q_1 v_1 \perp)$  and  $h(\perp u_2 q_2 v_2 \perp)$  the length might be the same. To overcome this problem we choose the fixed rate of compression  $2\mu$ , and simulate  $2\mu$  steps of  $M$  through a single application of a rule of  $R$ . If  $\perp u_1 q_1 v_1 \perp \vdash_M^{2\mu} \perp u_2 q_2 v_2 \perp$ , then  $|h(\perp u_1 q_1 v_1 \perp)| - 2\mu = |h(\perp u_2 q_2 v_2 \perp)|$ , and hence, if  $\gamma_1$  and  $\gamma_2$  are the compressed forms of  $h(\perp u_1 q_1 v_1 \perp)$  and  $h(\perp u_2 q_2 v_2 \perp)$ , respectively, then  $|\gamma_1| - 1 = \frac{|h(\perp u_1 q_1 v_1 \perp)| - 2\mu}{2\mu} = \frac{|h(\perp u_2 q_2 v_2 \perp)|}{2\mu} = |\gamma_2|$ .

To perform this construction we first determine the alphabet  $\Delta$ . Let  $\overline{\Gamma} \cup \{\overline{\#}\}$  be a new alphabet that is in 1-to-1 correspondence to  $\Gamma \cup \{\#\}$ , and let  $\overline{\phantom{x}} : \Gamma \cup \{\#\} \rightarrow \overline{\Gamma} \cup \{\overline{\#}\}$  denote this correspondence. Further, define four new alphabets as follows:

$$\begin{aligned}
 A_{\leq} &:= \{a_w \mid w \in (\Gamma \cup \{\#\})^* \text{ and } 1 \leq |w| \leq \mu\}, \\
 \overline{A} &:= \{a_w \mid w \in (\Gamma \cup \{\#\})^* \text{ and } |w| = 2\mu\}, \\
 \overline{A} &:= \{a_{\overline{w}} \mid \overline{w} \in (\overline{\Gamma} \cup \{\overline{\#}\})^* \text{ and } |\overline{w}| = 2\mu\}, \text{ and} \\
 A_Q &:= \{a_{\overline{u}qv} \mid \overline{u} \in (\overline{\Gamma} \cup \{\overline{\#}\})^*, q \in Q, v \in (\Gamma \cup \{\#\})^* \text{ and } |\overline{u}qv| = 2\mu\}.
 \end{aligned}$$

Thus, each letter  $a_w \in A_{\leq} \cup A \cup \overline{A} \cup A_Q$  represents a string  $w$  of length at most  $2\mu$ . Finally, we take  $\Delta := \Sigma \cup \{q_0, \perp, \perp, Y, N\} \cup A_{\leq} \cup A \cup \overline{A} \cup A_Q$ , where we assume that all the subalphabets displayed are pairwise disjoint.

To simplify the following considerations we define a morphism

$$\pi : (A_{\leq} \cup A \cup \overline{A} \cup A_Q)^* \rightarrow (\Gamma \cup Q \cup \{\#\})^*$$



through the following mapping:

$$a \mapsto \begin{cases} w, & \text{if } a = a_w \in A_{\leq} \cup A, \\ w, & \text{if } a = a_{\overline{w}} \in \overline{A}, \\ uqv, & \text{if } a = a_{\overline{u}qv} \in A_Q. \end{cases}$$

Thus,  $\pi$  replaces each letter  $a \in A_{\leq} \cup A \cup \overline{A} \cup A_Q$  by the string it represents, where in addition each factor  $\overline{u} \in (\overline{T} \cup \{\#\})^+$  is replaced by the corresponding string  $u \in (T \cup \{\#\})^+$ .

The string-rewriting system  $R$  will consist of four subsystems  $R_0$ ,  $R_1$ ,  $R_2$ , and  $R_3$ .

(0) The subsystem  $R_0$  is used to take care of those inputs  $w \in \Sigma^*$  for the sDTPDA  $M$  that are short:

$$R_0 := \{\overline{\perp}q_0w\perp \rightarrow Y \mid w \in \Sigma^*, \varphi(w\perp) \leq 4\mu, \text{ and } w \in L\} \\ \cup \{\overline{\perp}q_0w\perp \rightarrow N \mid w \in \Sigma^*, \varphi(w\perp) \leq 4\mu, \text{ and } w \notin L\}.$$

Obviously,  $R_0$  is a finite system containing only length-reducing rules, and there are no non-trivial overlaps between the left-hand sides of the rules of  $R_0$ .

(1) The subsystem  $R_1$  transforms the description  $\overline{\perp}q_0w\perp$  of an initial configuration  $\perp q_0w\perp$  of  $M$  into a compressed form  $c \in \overline{A}^* \cdot A_Q \cdot A^*$ , if  $w$  is sufficiently long. It consists of three parts.

$$(1.1) \ R_{1,1} := \{w\perp \rightarrow \alpha'\alpha_1\alpha_2 \mid w = av \in \Sigma^* \text{ for some } a \in \Sigma \text{ such that} \\ \varphi(v\perp) \leq 4\mu < \varphi(w\perp) \leq 5\mu, \alpha' \in A_{\leq}, \text{ and } \alpha_1, \alpha_2 \in A \text{ satisfying} \\ \pi(\alpha'\alpha_1\alpha_2) = h(w\perp)\}.$$

Since  $4\mu < \varphi(w\perp) \leq (|w| + 1) \cdot \mu$ , we see that  $|w| > 3$ . Hence,  $R_{1,1}$  is a finite system of length-reducing rules. The given weight restrictions for  $w\perp$  imply that the left-hand side of no rule of  $R_{1,1}$  is a proper suffix of the left-hand side of any other rule of  $R_{1,1}$ . Further, the right-hand side  $\alpha'\alpha_1\alpha_2$  of a rule of  $R_{1,1}$  is uniquely determined by the left-hand side, since the morphism  $h$  is injective. Hence, there are no non-trivial overlaps between the left-hand sides of the rules of  $R_{1,1}$ .

$$(1.2) \ R_{1,2} := \{w\alpha'_1 \rightarrow \alpha'_2\alpha \mid w = av \in \Sigma^* \text{ for some } a \in \Sigma, \alpha'_1, \alpha'_2 \in A_{\leq}, \\ \text{and } \alpha \in A \text{ such that } |h(v)\pi(\alpha'_1)| \leq 2\mu < |h(w)\pi(\alpha'_1)| \leq 3\mu \\ \text{and } \pi(\alpha'_2\alpha) = h(w)\pi(\alpha'_1)\}.$$

Since  $|\pi(\alpha'_1)| \leq \mu$ ,  $2\mu < |h(w)\pi(\alpha'_1)| = \varphi(w) + |\pi(\alpha'_1)|$  implies that  $\varphi(w) > \mu$ , which in turn yields  $|w| \geq 2$ . Hence,  $R_{1,2}$  is a finite system containing only length-reducing rules. As above it follows that there are no non-trivial overlaps between the left-hand sides of the rules of  $R_{1,2}$ .

(1.3) Working from right to left the rules of the subsystems  $R_{1,1}$  and  $R_{1,2}$  replace suffixes  $v\perp$  of  $\overline{\perp}q_0w\perp$  by the compressed form  $c \in A_{\leq} \cdot A^*$  of  $h(v\perp)$ . The subsystem  $R_{1,3}$  will be used to replace the remaining prefix  $\overline{\perp}q_0u$  such that the resulting string belongs to  $\overline{A}^* \cdot A_Q \cdot A^*$ , that is, it is the compressed

form of a string  $x \in (\Gamma \cup Q)^*$  satisfying  $|h(x)| \equiv 0 \pmod{2\mu}$ . Unfortunately, the initial configuration  $\perp q_0 w \perp$  may not satisfy this requirement. Therefore, if  $|h(\perp q_0 w \perp)| \equiv r \pmod{2\mu}$  for some  $r \in \{1, \dots, 2\mu-1\}$ , then instead of compressing this initial configuration, we compress the configuration  $\perp u q v \perp$  that is obtained from  $\perp q_0 w \perp$  after  $r$  steps of  $M$ . Then  $|h(\perp u q v \perp)| = |h(\perp q_0 w \perp)| - r \equiv 0 \pmod{2\mu}$ , and hence,  $h(\perp u q v \perp)$  can be encoded through a string  $c \in \bar{A}^* \cdot A_Q \cdot A^*$  such that  $\pi(c) = h(\perp u q v \perp)$ .

In each step the sDTPDA  $M$  can remove at most one symbol from the top of its second pushdown store. Thus, the first  $2\mu - 1$  steps of the computation of  $M$  on input  $w$  depend only on the prefix  $u$  of  $w$  of length  $2\mu - 1$ . Hence, the rules of  $R_{1,3}$  will encode all computations of  $M$  of this form.

$$\begin{aligned}
 R_{1,3} := \{ & \bar{\perp} q_0 w \alpha' \alpha_1 \cdots \alpha_n \rightarrow \beta_1 \cdots \beta_m \mid w \in \Sigma^*, \alpha' \in A_{\leq}, \alpha_1, \dots, \alpha_n \in A \\
 & \text{such that } |h(w)\pi(\alpha')| \leq 2\mu, 2 \leq n \leq \mu, \text{ where } n < \mu \\
 & \text{implies that } \pi(\alpha' \alpha_1 \cdots \alpha_n) \in ((\Gamma \setminus \{\perp\}) \cup \{\#\})^* \cdot h(\perp), \\
 & \text{and } \beta_1, \dots, \beta_m \in \bar{A} \cup A_Q \cup A \text{ satisfy the following conditions:} \\
 & \text{(i) } \beta_1 \cdots \beta_m \in \bar{A}^* \cdot A_Q \cdot A^*, \\
 & \text{(ii) } h(w)\pi(\alpha' \alpha_1 \cdots \alpha_n) = h(v)x \text{ for some } v \in (\Gamma \setminus \{\perp\})^* \cdot \{\perp, \varepsilon\} \\
 & \quad \text{and } x \in (\Gamma \cup \{\#\})^* \text{ satisfying } |x| < \mu, \text{ and} \\
 & \text{(iii) } \pi(\beta_1 \cdots \beta_m) = h(u_1 q_1 v_1)x \text{ for some } u_1, v_1 \in \Gamma^* \text{ and } q_1 \in Q \\
 & \quad \text{such that } \perp q_0 v \vdash_M^r u_1 q_1 v_1, \text{ where } r \in \{0, 1, \dots, 2\mu - 1\} \\
 & \quad \text{satisfies } |h(\perp q_0 w)\pi(\alpha')| \equiv r \pmod{2\mu}.
 \end{aligned}$$

If  $(\bar{\perp} q_0 w \alpha' \alpha_1 \cdots \alpha_n \rightarrow \beta_1 \cdots \beta_m) \in R_{1,3}$ , then  $m \in \{n, n+1, n+2\}$ . Hence,  $R_{1,3}$  is a finite system of length-reducing rules. It can easily be checked that there are no non-trivial overlaps between the left-hand sides of the rules of  $R_{1,3}$ .

The subsystem  $R_1$  is now taken as  $R_1 := R_{1,1} \cup R_{1,2} \cup R_{1,3}$ . From the definitions given it follows immediately that there are no non-trivial overlaps between the left-hand sides of the rules of  $R_1$ .

(2) The subsystem  $R_2$  simulates the computations of the sDTPDA  $M$  on strings that represent compressed forms of configurations. Each application of a rule of  $R_2$  simulates  $2\mu$  steps of  $M$ .

$$\begin{aligned}
 R_2 := \{ & \alpha_1 \cdots \alpha_n \gamma \alpha_{n+1} \cdots \alpha_{n+m} \rightarrow \beta_1 \cdots \beta_{n+m} \mid \alpha_1, \dots, \alpha_n \in \bar{A}, \gamma \in A_Q, \\
 & \alpha_{n+1}, \dots, \alpha_{n+m} \in A \text{ such that } n, m \leq \mu + 1, \text{ where} \\
 & 1 \leq n \leq \mu \text{ implies that } \pi(\alpha_1) \text{ has prefix } h(\perp), \\
 & n = 0 \text{ implies that } \pi(\gamma) \text{ has prefix } h(\perp) \text{ and } m \geq 2, \\
 & 1 \leq m \leq \mu \text{ implies that } \pi(\alpha_{n+m}) \text{ has suffix } h(\perp), \text{ and} \\
 & m = 0 \text{ implies that } \pi(\gamma) \text{ has suffix } h(\perp) \text{ and } n \geq 2, \\
 & \beta_1, \dots, \beta_{n+m} \in \bar{A} \cup A_Q \cup A \text{ such that } \beta_1 \cdots \beta_{n+m} \in \bar{A}^* \cdot A_Q \cdot A^*, \\
 & \pi(\alpha_1 \cdots \alpha_n \gamma \alpha_{n+1} \cdots \alpha_{n+m}) = x_1 h(u q v) x_2 \text{ for some } u, v \in \Gamma^*, q \in Q, \\
 & x_1 \in \{\#\}^*, x_2 \in \Gamma \cdot \{\#\}^*, |x_1|, |x_2| < \mu, x_2 \notin h(\Gamma), \\
 & \text{and } \pi(\beta_1 \cdots \beta_{n+m}) = x_1 h(u_1 q_1 v_1) x_2 \text{ for some } u_1, v_1 \in \Gamma^*, q_1 \in Q, \\
 & \text{such that } u q v \text{ and } u_1 q_1 v_1 \text{ are valid subconfigurations of } M \\
 & \text{satisfying } u q v \vdash_M^{2\mu} u_1 q_1 v_1 \}.
 \end{aligned}$$

The conditions on the integers  $n$  and  $m$  imply that  $n + m \geq 2$ . Further, all rules of  $R_2$  are obviously length-reducing. Since  $uqv$  and  $u_1q_1v_1$  must be valid subconfigurations of  $M$ ,  $\perp$  can occur at most as the first and/or the last letter. Hence, the left-hand side of no rule of  $R_2$  is contained in the left-hand side of another rule of  $R_2$ . Finally, the right-hand side of a rule of  $R_2$  is uniquely determined by its left-hand side. Thus, there are no non-trivial overlaps between the left-hand side of the rules of  $R_2$ .

(3) The subsystem  $R_3$  ends the simulation of computations of  $M$ .

$$R_3 := \{\alpha_1\alpha_2 \rightarrow Y \mid \alpha_1, \alpha_2 \in \bar{A} \cup A_Q \cup A, \alpha_1\alpha_2 \in \bar{A}^* \cdot A_Q \cdot A^*, \pi(\alpha_1\alpha_2) = h(\perp uqv \perp) \text{ for some } u, v \in \Gamma^* \text{ and } q \in Q, \text{ and } \perp uqv \perp \vdash_M^* q_f\} \\ \cup \{\alpha_1\alpha_2 \rightarrow N \mid \alpha_1, \alpha_2 \in \bar{A} \cup A_Q \cup A, \alpha_1\alpha_2 \in \bar{A}^* \cdot A_Q \cdot A^*, \pi(\alpha_1\alpha_2) = h(\perp uqv \perp) \text{ for some } u, v \in \Gamma^* \text{ and } q \in Q, \text{ and } \perp uqv \perp \vdash_M^* \perp q_f\}.$$

Obviously,  $R_3$  is a finite length-reducing system, and there are no non-trivial overlaps between the left-hand side of the rules of  $R_3$ .

Finally, we take  $R := R_0 \cup R_1 \cup R_2 \cup R_3$ . Then  $R$  is indeed a finite string-rewriting system that contains length-reducing rules only. It is easily verified that there are no non-trivial overlaps between the left-hand sides of the rules of  $R$ . Hence, we see that  $R$  is also confluent.

It remains to prove the following statements for all  $w \in \Sigma^*$ :

- (i) If  $w \in L$ , then  $\bar{\perp}q_0w\perp \rightarrow_R^* Y$ .
- (ii) If  $w \notin L$ , then  $\bar{\perp}q_0w\perp \rightarrow_R^* N$ .

These statements show that the system  $R$ , together with the strings  $t_1 := \bar{\perp}q_0$  and  $t_2 := \perp$  and the letters  $Y$  and  $N$ , witnesses the fact that  $L$  is a Church-Rosser decidable language. The proof of the statements above will be divided into several claims and their proofs. The first one follows immediately from the choice of the subsystem  $R_0$ .

**Claim 1.** For all  $w \in \Sigma^*$  satisfying  $\varphi(w) \leq 4\mu - \varphi(\perp)$  the statements (i) and (ii) hold.

Hence, for the following considerations we can assume that the string  $w \in \Sigma^*$  satisfies  $\varphi(w) > 4\mu - \varphi(\perp)$ , that is,  $\varphi(w\perp) > 4\mu$ .

**Claim 2.** Let  $w \in \Sigma^*$  such that  $\varphi(w\perp) > 4\mu$ , and let  $r \in \{0, 1, \dots, 2\mu - 1\}$  such that  $\varphi(\perp q_0w\perp) = k \cdot 2\mu + r$  for some  $k \in \mathbb{N}$ . Then there exist  $\alpha_1, \dots, \alpha_k \in \bar{A} \cup A_Q \cup A$  satisfying the following conditions:

- (i)  $\alpha_1\alpha_2 \cdots \alpha_k \in \bar{A}^* \cdot A_Q \cdot A^*$ ,
- (ii)  $\pi(\alpha_1 \cdots \alpha_k) = h(\perp uqv \perp)$  for some configuration  $\perp uqv \perp$  of  $M$ ,  
where  $\perp q_0w\perp \vdash_M^r \perp uqv \perp$ , and
- (iii)  $\bar{\perp}q_0w\perp \rightarrow_{R_1}^* \alpha_1 \cdots \alpha_k$ .

*Proof.* Let  $w \in \Sigma^*$  satisfy  $\varphi(w\perp) > 4\mu$ , and let  $k \in \mathbb{N}$  and  $r \in \{0, 1, \dots, 2\mu - 1\}$  such that  $\varphi(\perp q_0w\perp) = k \cdot 2\mu + r > 4\mu$ . The computation of  $M$  starting from

the initial configuration  $\perp q_0 w \perp$  either ends with the accepting configuration  $q_f$  of weight  $\varphi(q_f) \leq \mu$  or with the non-accepting configuration  $\perp q_f$  of weight  $\varphi(\perp q_f) \leq 2\mu$ . Hence, this computation consists of more than  $2\mu$  steps. Thus, there is a (uniquely determined) configuration  $\perp uqv \perp$  of  $M$  such that  $\perp q_0 w \perp \vdash_M^r \perp uqv \perp$ . Since  $\varphi(\perp uqv \perp) = \varphi(\perp q_0 w \perp) - r = k \cdot 2\mu$ , there exist  $\alpha_1, \dots, \alpha_k \in \overline{A} \cup A_Q \cup A$  such that  $\alpha_1 \alpha_2 \cdots \alpha_k \in \overline{A}^* \cdot A_Q \cdot A^*$  and  $\pi(\alpha_1 \cdots \alpha_k) = h(\perp uqv \perp)$ . It follows easily from the definition of the rules of the system  $R_1$  that  $\perp q_0 w \perp \rightarrow_{R_{1,1}} w_1 \rightarrow_{R_{1,2}}^* w_2 \rightarrow_{R_{1,3}} \alpha_1 \alpha_2 \cdots \alpha_k$  holds for some strings  $w_1$  and  $w_2$ .  $\square$

**Claim 3.** Let  $\perp uqv \perp$  be a configuration of  $M$  such that  $\varphi(\perp uqv \perp) = s \cdot 2\mu$  for some  $s \geq 3$ , and let  $\alpha_1, \dots, \alpha_s \in \overline{A} \cup A_Q \cup A$  such that  $\alpha_1 \cdots \alpha_s \in \overline{A}^* \cdot A_Q \cdot A^*$  and  $\pi(\alpha_1 \cdots \alpha_s) = h(\perp uqv \perp)$ . If  $\perp uqv \perp$  is reachable from an initial configuration, then there exist a configuration  $\perp u_1 q_1 v_1 \perp$  of  $M$  and letters  $\beta_1, \dots, \beta_{s-1} \in \overline{A} \cup A_Q \cup A$  such that the following conditions are satisfied:

- (i)  $\beta_1 \beta_2 \cdots \beta_{s-1} \in \overline{A}^* \cdot A_Q \cdot A^*$ ,
- (ii)  $\pi(\beta_1 \beta_2 \cdots \beta_{s-1}) = h(\perp u_1 q_1 v_1 \perp)$ ,
- (iii)  $\perp uqv \perp \vdash_M^{2\mu} \perp u_1 q_1 v_1 \perp$ , and
- (iv)  $\alpha_1 \alpha_2 \cdots \alpha_s \rightarrow_{R_2} \beta_1 \beta_2 \cdots \beta_{s-1}$ .

*Proof.* Let  $\perp uqv \perp$  be a configuration of  $M$  such that  $\varphi(\perp uqv \perp) = s \cdot 2\mu$  for some  $s \geq 3$ . If  $\perp uqv \perp$  is reachable from some initial configuration, that is,  $\perp q_0 w \perp \vdash_M^r \perp uqv \perp$  for some  $w \in \Sigma^*$ , then  $\perp uqv \perp \vdash_M^* q_f$  or  $\perp uqv \perp \vdash_M^* \perp q_f$ , depending on whether  $w \in L$  or  $w \notin L$ , respectively. Since the weight of the actual configuration decreases by 1 in each step, we see that there exists a unique configuration  $\perp u_1 q_1 v_1 \perp$  such that  $\perp uqv \perp \vdash_M^{2\mu} \perp u_1 q_1 v_1 \perp$  and  $\varphi(\perp u_1 q_1 v_1 \perp) = \varphi(\perp uqv \perp) - 2\mu = (s-1) \cdot 2\mu$ . Hence, there exist (uniquely determined)  $\beta_1, \beta_2, \dots, \beta_{s-1} \in \overline{A} \cup A_Q \cup A$  satisfying  $\beta_1 \beta_2 \cdots \beta_{s-1} \in \overline{A}^* \cdot A_Q \cdot A^*$  and  $\pi(\beta_1 \beta_2 \cdots \beta_{s-1}) = h(\perp u_1 q_1 v_1 \perp)$ .

During the computation  $\perp uqv \perp \vdash_M^{2\mu} \perp u_1 q_1 v_1 \perp$  a suffix  $u'$  of  $u$  and a prefix  $v'$  of  $v$  are involved that satisfy  $|u'|, |v'| \leq 2\mu$ . Hence, this computation can be described completely by using a window of length  $2\mu + 1 + 2\mu = 4\mu + 1$  that is placed on  $\perp uqv \perp$  in such a way that the state symbol  $q$  appears in the middle. The corresponding section of  $h(\perp uqv \perp)$  is contained in a substring  $\alpha'_1 \cdots \alpha'_n \gamma' \alpha'_{n+1} \cdots \alpha'_{n+m} \in \overline{A}^* \cdot A_Q \cdot A^*$  of  $\alpha_1 \alpha_2 \cdots \alpha_s$  satisfying  $n, m \leq \mu + 1$ . From the definition of the subsystem  $R_2$  we see that each rule of  $R_2$  just simulates  $2\mu$  steps of  $M$  on a substring of this form. Hence, it follows that  $\alpha_1 \alpha_2 \cdots \alpha_s \rightarrow_{R_2} \beta_1 \beta_2 \cdots \beta_{s-1}$  holds.  $\square$

**Claim 4.** Let  $\perp uqv \perp$  be a configuration of  $M$  such that  $\varphi(\perp uqv \perp) = 4\mu$ , and let  $\alpha_1, \alpha_2 \in \overline{A} \cup A_Q \cup A$  such that  $\alpha_1 \alpha_2 \in \overline{A}^* \cdot A_Q \cdot A^*$  and  $\pi(\alpha_1 \alpha_2) = h(\perp uqv \perp)$ . If  $\perp uqv \perp$  is reachable from an initial configuration, then either  $\alpha_1 \alpha_2 \rightarrow_{R_3} Y$  or  $\alpha_1 \alpha_2 \rightarrow_{R_3} N$ .

*Proof.* Let  $\perp uqv \perp$  be a configuration of  $M$  such that  $\varphi(\perp uqv \perp) = 4\mu$ , and let  $\alpha_1, \alpha_2 \in \overline{A} \cup A_Q \cup A$  such that  $\alpha_1 \alpha_2 \in \overline{A}^* \cdot A_Q \cdot A^*$  and  $\pi(\alpha_1 \alpha_2) = h(\perp uqv \perp)$ . If  $\perp uqv \perp$  is reachable from some initial configuration, then  $\perp q_0 w \perp \vdash_M^* \perp uqv \perp$  for some  $w \in \Sigma^*$ . If  $w \in L$ , then  $\perp uqv \perp \vdash_M^* q_f$ , and if  $w \notin L$ , then  $\perp uqv \perp \vdash_M^* \perp q_f$ . Thus, either  $(\alpha_1 \alpha_2 \rightarrow Y) \in R_3$  or  $(\alpha_1 \alpha_2 \rightarrow N) \in R_3$ .  $\square$

We now verify that  $R$  does indeed witness the fact that  $L$  is a Church-Rosser decidable language. Let  $w \in \Sigma^*$ . If  $\varphi(w) \leq 4\mu - \varphi(\perp)$ , then we see from Claim 1 that  $\perp q_0 w \perp \rightarrow_R Y$  if  $w \in L$ , and  $\perp q_0 w \perp \rightarrow_R N$ , if  $w \notin L$ . Assume therefore that  $\varphi(w) > 4\mu - \varphi(\perp)$ . Then by Claim 2 there exist a configuration  $\perp u_1 q_1 v_1 \perp$  of  $M$  and  $\alpha_1, \alpha_2, \dots, \alpha_k \in \overline{A} \cup A_Q \cup A$  such that

- (i)  $\alpha_1 \alpha_2 \cdots \alpha_k \in \overline{A}^* \cdot A_Q \cdot A^*$ ,
- (ii)  $\pi(\alpha_1 \alpha_2 \cdots \alpha_k) = h(\perp u_1 q_1 v_1 \perp)$ ,
- (iii)  $\perp q_0 w \perp \vdash_M^* \perp u_1 q_1 v_1 \perp$ , and
- (iv)  $\perp q_0 w \perp \rightarrow_R^* \alpha_1 \cdots \alpha_k$ .

If  $k > 2$ , then Claim 3 applies. Hence, there are configurations  $\perp u_i q_i v_i \perp$  of  $M$  and strings  $\delta_i \in \overline{A}^* \cdot A_Q \cdot A^*$ ,  $i = 2, \dots, k-1$ , such that  $\perp u_{i-1} q_{i-1} v_{i-1} \perp \vdash_M^{2\mu} \perp u_i q_i v_i \perp$ ,  $\pi(\delta_i) = h(\perp u_i q_i v_i \perp)$ ,  $\alpha_1 \cdots \alpha_k \rightarrow_R \delta_2 \rightarrow_R \dots \rightarrow_R \delta_{k-1}$ , and  $|\delta_i| = k - i + 1$  for all  $i = 2, \dots, k-1$ . Finally,  $|\delta_{k-1}| = 2$  implies that  $\delta_{k-1} \rightarrow_R Y$  or  $\delta_{k-1} \rightarrow_R N$  by Claim 4. From the definition of  $R_3$  we see that the former is the case if and only if  $w \in L$ . Thus, for  $w \in L$ , we have  $\perp q_0 w \perp \rightarrow_R^* \alpha_1 \cdots \alpha_k \rightarrow_R \dots \rightarrow_R \delta_{k-1} \rightarrow_R Y$ , and for  $w \notin L$ , we have  $\perp q_0 w \perp \rightarrow_R^* \alpha_1 \cdots \alpha_k \rightarrow_R \dots \rightarrow_R \delta_{k-1} \rightarrow_R N$ . This completes the proof of Theorem 5.  $\square$

From Theorem 5 we obtain our main result.

**Corollary 6.** *The three language classes CRDL, CRL, and GCRL coincide.*

Thus, the Church-Rosser languages are indeed the deterministic variants of the growing context-sensitive languages.

## 5 Closure Properties

In this section we summarize the known closure and non-closure properties of the class CRL and we prove two new non-closure properties, which, however, were already announced by Buntrock and Otto [7].

From the definition of the class CRDL we immediately obtain the following result.

**Proposition 7.** *The class of Church-Rosser languages is closed under complementation, that is, if  $L \subseteq \Sigma^*$  is a Church-Rosser language, then so is the language  $\overline{L} := \Sigma^* \setminus L$ .*

From the characterization of the class GCRL through the shrinking DTPDA we can conclude the following closure properties.

**Proposition 8.**

- (a) *The class CRL is closed under intersection with regular languages, that is, if  $L \in \text{CRL}$  and  $L_1$  is a regular language, then  $L \cap L_1 \in \text{CRL}$ .*
- (b) *The class CRL is closed under inverse morphisms, that is, if  $L \subseteq \Sigma^*$  is in CRL and  $h : \Delta^* \rightarrow \Sigma^*$  is a morphism, then  $h^{-1}(L) \in \text{CRL}$ .*

Finally, from [11] we recall the following closure properties.

**Proposition 9.**

- (a) *CRL is closed under reversal, that is, if  $L$  is a Church-Rosser language, then so is the language  $L^\sim := \{w^\sim \mid w \in L\}$ .*
- (b) *CRL is closed under left quotient and right quotient with a single string, that is, if  $L \subseteq \Sigma^*$  is a Church-Rosser language and  $z \in \Sigma^*$ , then  $L/\{z\} = \{w \in \Sigma^* \mid wz \in L\}$  and  $\{z\} \setminus L := \{w \in \Sigma^* \mid zw \in L\}$  are Church-Rosser languages, too.*

In [12] it is shown that the class CRL is a basis for the recursively enumerable languages. Further, it is shown by Buntrock in [4] that the closure of the class GCRL (= CRL) under  $\varepsilon$ -free morphisms yields the class GCSL. Hence, we obtain the following non-closure properties.

**Proposition 10.** *The class CRL is neither closed under projections nor under  $\varepsilon$ -free morphisms.*

The Gladkij language  $L_{G1} := \{w\phi w^\sim \phi w \mid w \in \{a, b\}^*\}$  is a context-sensitive language that is not growing context-sensitive [9, 1, 7]. Now  $L_{G1}$  can be written as  $L_{G1} = L_1 \cap L_2$ , where  $L_1 := \{w\phi w^\sim \phi z \mid w, z \in \{a, b\}^*\}$  and  $L_2 := \{w\phi z\phi z^\sim \mid w, z \in \{a, b\}^*\}$ . Obviously,  $L_1$  and  $L_2$  are both deterministic context-free, and hence, they are both Church-Rosser languages. Since  $L_1 \cap L_2 \notin \text{GCSL}$ , we have  $L_1 \cap L_2 \notin \text{CRL}$ . This shows the following.

**Proposition 11.** *The class CRL is neither closed under intersection nor under union.*

## 6 Conclusion

We have shown that the three language classes CRDL and CRL of [11] and GCRL of [7] coincide. Because of the characterization of the latter class through the deterministic variant of the shrinking TPDA [7] this class of languages can be considered as the class of ‘deterministic growing context-sensitive languages’. Based on these characterizations we have obtained some closure properties and some non-closure properties for the class of Church-Rosser languages. However, many questions regarding closure and non-closure properties remain open. Also it

remains the question of whether or not the language  $L_0 := \{ww^\sim \mid w \in \{a,b\}^*\}$  is a Church-Rosser language.

Finally, based on the fact that the classes CFL and CRL are incomparable under set inclusion, we obtain the following undecidability result from McNaughton et al [11].

**Proposition 12.**

- (a) *The emptiness and the finiteness problems for Church-Rosser languages are undecidable in general.*
- (b) *It is undecidable in general whether a given context-free language is a Church-Rosser language.*
- (c) *It is undecidable in general whether a given Church-Rosser language is context-free.*

## References

1. R.V. Book. *Grammars with Time Functions*. PhD thesis, Harvard University, Cambridge, Massachusetts, February 1969.
2. R.V. Book. Confluent and other types of Thue systems. *J. Association Computing Machinery*, 29:171–182, 1982.
3. R.V. Book and F. Otto. *String-Rewriting Systems*. Springer-Verlag, New York, 1993.
4. G. Buntrock. *Wachsende kontext-sensitive Sprachen*. Habilitationsschrift, Fakultät für Mathematik und Informatik, Universität Würzburg, July 1996.
5. G. Buntrock and K. Loryś. On growing context-sensitive languages. In W. Kuich, editor, *Proc. of 19th ICALP*, Lecture Notes in Computer Science 623, pages 77–88. Springer-Verlag, Berlin, 1992.
6. G. Buntrock and K. Loryś. The variable membership problem: Succinctness versus complexity. In P. Enjalbert, E.W. Mayr, and K.W. Wagner, editors, *Proc. of 11th STACS*, Lecture Notes in Computer Science 775, pages 595–606. Springer-Verlag, Berlin, 1994.
7. G. Buntrock and F. Otto. Growing context-sensitive languages and Church-Rosser languages. In E.W. Mayr and C. Puech, editors, *Proc. of 12th STACS*, Lecture Notes in Computer Science 900, pages 313–324. Springer-Verlag, Berlin, 1995.
8. E. Dahlhaus and M. Warmuth. Membership for growing context-sensitive grammars is polynomial. *J. Computer System Sciences*, 33:456–472, 1986.
9. A.W. Gladkij. On the complexity of derivations for context-sensitive grammars. *Algebra i Logika Sem.*, 3:29–44, 1964. In Russian.
10. J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, Reading, M.A., 1979.
11. R. McNaughton, P. Narendran, and F. Otto. Church-Rosser Thue systems and formal languages. *J. Association Computing Machinery*, 35:324–344, 1988.
12. F. Otto, M. Katsura, and Y. Kobayashi. Cross-sections for finitely presented monoids with decidable word problems. In H. Comon, editor, *Rewriting Techniques and Applications*, Lecture Notes in Computer Science 1232, pages 53–67. Springer-Verlag, Berlin, 1997.