



**HAL**  
open science

# Modular termination of prefix-constrained term rewrite systems

Nirina Andrianarivelo, Pierre Réty

► **To cite this version:**

Nirina Andrianarivelo, Pierre Réty. Modular termination of prefix-constrained term rewrite systems. Information Processing Letters, 2022, 10.1016/j.ipl.2021.106207 . hal-03356610

**HAL Id: hal-03356610**

**<https://inria.hal.science/hal-03356610v1>**

Submitted on 16 Oct 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

# Modular Termination of Prefix-Constrained Term Rewrite Systems

Nirina Andrianarivelo

Pierre Réty

LIFO - Université d'Orléans, B.P. 6759, 45067 Orléans cedex 2, France  
E-mail : {Nirina.Andrianarivelo, Pierre.Rety}@univ-orleans.fr

**Keywords :** prefix-constrained rewriting, modularity of termination.

## Abstract

We study the modularity of termination for prefix-constrained rewrite systems, and extend the known results of standard and context-sensitive rewriting.

## 1. Introduction

Term rewriting is a rule-based formalism that can be used to study properties of functional programs, security protocols, musical rhythmic patterns. More generally, it provides a finite abstraction of a system whose configurations are represented by terms. In this framework, and also to ensure the termination of rewrite computations, it is often necessary to restrict the possible rewrite positions, using strategies, or by allowing only some redex positions. In context-sensitive rewriting [7], some arguments of a function symbol may be defined as being non-reducible. Prefix-constrained rewriting [6] is an extension of context-sensitive rewriting, where potentially reducible positions are defined by regular string languages that indicate the allowed prefixes.

A modular approach can facilitate the proof of termination of rewrite systems. Termination is modular (under some assumptions) if the termination of rewrite systems  $R_1, R_2$  implies the termination of  $R_1 \cup R_2$ . Modular termination of standard rewrite systems has been studied in [11, 8, 12, 9, 3, 4, 2, 10], and in [5] for context-sensitive rewrite systems. In this paper, we extend the results of [10] and [5] to prefix-constrained rewrite systems. A big difficulty comes from the fact that prefix-constrained rewriting (denoted  $\hookrightarrow$ ) is not closed when removing a context, i.e.  $C[l] \hookrightarrow_{[l \rightarrow r]} C[r] \not\Rightarrow l \hookrightarrow r$ , whereas it holds for standard and context-sensitive rewriting. Proofs of [10, 5] are based on this property.

The term  $t = f(g(a, b))$  is reducible at position 1.2 into  $f(g(a, c))$  by the prefix-constrained rule  $\langle f, 1 \rangle . \langle g, 2 \rangle : b \rightarrow c$ , since the *path* going from the root of  $t$  to  $b$  is  $\langle f, 1 \rangle . \langle g, 2 \rangle$ , which implies that the prefix-constraint of the rule is satisfied. On the other hand,  $t$  cannot be reduced at position 1.1 by the rule  $\langle f, 1 \rangle . \langle g, 2 \rangle : a \rightarrow c$ , since the path going from the root of  $t$  to  $a$  is  $\langle f, 1 \rangle . \langle g, 1 \rangle$ . The prefix constraint is a regular string language, expressed by a regular expression.

We get the following results, given at the end of the paper. If  $R_1$  and  $R_2$  are *totally disjoint*, i.e. the rules (including constraints) of  $R_1$  and  $R_2$  do

not share function symbols, then termination is automatically modular (Theorem 27). On the other hand, if  $R_1$  and  $R_2$  are only *weakly disjoint*, i.e. the rules (without including constraints) of  $R_1$  and  $R_2$  do not share function symbols, then termination is modular under some restrictions (Theorem 28). As in [10], the restrictions use the notions of duplicating rule and collapsing rule. However, the notion of duplicating rule is more sophisticated since it needs to take constraints into account. As for context-sensitive rewriting, a linear rule like  $\epsilon : f(x) \rightarrow g(x)$  may be duplicating if for example the position of  $x$  in  $f(x)$  is not reducible according to the constraints, whereas it is reducible in  $g(x)$ . However, contrary to the context-sensitive case, this property is not local, i.e. it also depends on the other rules of the rewrite system (Definitions 21 and 22).

**Example 1.**  $R_1 = \{\langle h, 1 \rangle : f(a, b, x) \rightarrow f(x, x, x)\}$ ,  $R_2 = \{\langle h, 1 \rangle.\langle f, 1 \rangle : g(x, y) \rightarrow x, \langle h, 1 \rangle.\langle f, 2 \rangle : g(x, y) \rightarrow y\}$ . In this example inspired from [13], the rule of  $R_1$  is duplicating and the rules of  $R_2$  are collapsing. Termination is not modular because  $R_1$  and  $R_2$  are terminating whereas  $R_1 \cup R_2$  is not :  $h(f(a, b, g(a, b))) \hookrightarrow h(f(g(a, b), g(a, b), g(a, b))) \hookrightarrow h(f(a, g(a, b), g(a, b))) \hookrightarrow h(f(a, b, g(a, b))) \hookrightarrow \dots$

Now let  $R_3 = \{\langle h, 1 \rangle : g(x, y) \rightarrow x, \langle h, 1 \rangle : g(x, y) \rightarrow y\}$ . In  $R_1 \cup R_3$ , the rule of  $R_1$  is not duplicating and  $h(f(a, b, g(a, b))) \hookrightarrow h(f(g(a, b), g(a, b), g(a, b)))$  stops here. In this case, termination is modular.

Another idea to prove modular termination could consist in transforming the prefix-constrained rewrite systems  $R_1, R_2$  into standard rewrite systems  $\mathcal{T}(R_1), \mathcal{T}(R_2)$  [1]. Then one could use the known results about modular termination of standard rewriting [10]. Unfortunately, it does not work since the additional symbols introduced in  $\mathcal{T}(R_1), \mathcal{T}(R_2)$  by the transformation should be common, i.e.  $\mathcal{T}(R_1)$  and  $\mathcal{T}(R_2)$  share some function symbols, and then are not disjoint.

## 2. Preliminaries

**Term and Substitution.** Consider a finite *signature*  $\Sigma$  and a set of variables  $X$ .  $T(\Sigma, X)$  denotes the set of terms over  $\Sigma \cup X$ . For a term  $t$ ,  $Var(t)$  is the set of variables of  $t$ ,  $Pos(t)$  is the set of positions of  $t$ ,  $posVar(t)$  is the set of variable positions of  $t$ , and  $\epsilon$  is the root position. For  $p \in Pos(t)$ ,  $t(p)$  is the symbol of  $\Sigma \cup X$  occurring at position  $p$  in  $t$ , and  $t|_p$  is the subterm of  $t$  at position  $p$ . For  $p, p' \in Pos(t)$ ,  $p \leq p'$  means that  $p$  is a prefix of  $p'$ , i.e.  $p$  occurs above  $p'$ , and  $p \parallel p' \Leftrightarrow \neg(p \leq p' \vee p' \leq p)$ . The term  $t[t']_p$  is obtained from  $t$  by replacing the subterm at position  $p$  by  $t'$ .

**Term Rewrite System (TRS).** A *rewrite rule* is an oriented pair of terms, written  $l \rightarrow r$  such that  $l$  is not a variable, and  $Var(r) \subseteq Var(l)$ . The rule is said *collapsing* if  $r$  is a variable. A *rewrite system*  $R$  is a finite set of rewrite rules. The *rewrite relation*  $\rightarrow_R$  is defined as follows:  $t \rightarrow_R t'$  if there exist a non-variable position  $p \in Pos(t)$ , a rule  $l \rightarrow r \in R$ , and a substitution  $\sigma$  s.t.  $t|_p = \sigma(l)$  and  $t' = t[\sigma(r)]_p$  (also denoted  $t \rightarrow_{[p]} t'$ ).

**String Language.** The set of all strings over the alphabet  $A$  is denoted by  $A^*$ , and  $\epsilon$  denotes the *empty string*. Symbol ‘.’ denotes the concatenation. In examples, regular string languages will be expressed by regular expressions.

**Prefix-Constrained Term Rewrite System (pCTRS) [6].** Consider the set of directions  $Dir(\Sigma) = \{\langle g, i \rangle \mid g \in \Sigma, 1 \leq i \leq ar(g)\}$  where  $ar(g)$  is the arity of  $g$ . For a term  $t = g(t_1, \dots, t_{ar(g)}) \in T(\Sigma)$  and a position  $p$ ,  $path(t, p) \in Dir(\Sigma)^*$  is defined recursively by:

$$\begin{aligned} path(g(t_1, \dots, t_{ar(g)}), \epsilon) &= \epsilon \\ path(g(t_1, \dots, t_{ar(g)}), i.p) &= \langle g, i \rangle . path(t_i, p) \text{ with } 1 \leq i \leq ar(g) \end{aligned}$$

A *prefix-constrained rewrite system* is a finite set  $R$  of rules of the form  $L : l \rightarrow r$  s.t.  $L \subseteq Dir(\Sigma)^*$  is a regular string language over  $Dir(\Sigma)$ , and  $l \rightarrow r$  is a standard rewrite rule.  $t$  is reduced into  $t'$  by a pCTRS  $R$ , denoted by  $t \hookrightarrow_R t'$ , if there exist a rule  $L : l \rightarrow r$  in  $R$ , a position  $p \in Pos(t)$  s.t.  $path(t, p) \in L$ , and a substitution  $\sigma$  s.t.  $t|_p = \sigma(l)$  and  $t' = t[\sigma(r)]_p$  (also denoted  $t \hookrightarrow_{[p, L: l \rightarrow r, \sigma]} t'$ ).

**Example 2.** Let  $\Sigma = \{f, g, a, b\}$  and  $R = \{(\langle f, 1 \rangle . \langle g, 2 \rangle)^* : a \rightarrow b\}$ . Let  $t = f(g(a, \mathbf{a}), a)$ . Note that  $t(1.2) = a$  (in bold in  $t$ ) and  $path(t, 1.2) = \langle f, 1 \rangle . \langle g, 2 \rangle \in (\langle f, 1 \rangle . \langle g, 2 \rangle)^*$ . Then this position can be reduced by prefix-constrained rewriting, i.e.  $t = f(g(a, \mathbf{a}), a) \hookrightarrow_R f(g(a, b), a)$ , whereas the other occurrences of  $a$  are not reducible.

### 3. Modular Termination of pCTRSs

**Definition 3.** Let  $Sym(t)$  denote the non-variable symbols of the term  $t$ , and  $root(t)$  denote its root symbol. The pCTRSs  $R_1, R_2$  are *weakly disjoint* if there are signatures  $\Sigma_1, \Sigma_2$  s.t.  $\Sigma = \Sigma_1 \cup \Sigma_2$  (i.e.  $\Sigma_1 \cap \Sigma_2 = \emptyset$ ), and for every rule  $L_1 : l_1 \rightarrow r_1$  of  $R_1$ ,  $Sym(l_1) \cup Sym(r_1) \subseteq \Sigma_1$ , and for every rule  $L_2 : l_2 \rightarrow r_2$  of  $R_2$ ,  $Sym(l_2) \cup Sym(r_2) \subseteq \Sigma_2$ . If in addition  $L_1 \subseteq Dir(\Sigma_1)^*$  and  $L_2 \subseteq Dir(\Sigma_2)^*$ , then  $R_1, R_2$  are *totally disjoint*. In the sequel, their union is denoted by  $R_1 \cup R_2$ .

In Example 1,  $R_1, R_2$  are weakly disjoint, as well as  $R_1, R_3$ .

Roughly speaking, within a term we need to consider homogenous layers. Within a layer, all function symbols belong to the same  $\Sigma_i$ .  $top(t)$  is the topmost layer of the term  $t$ , and  $rank(t)$  denotes the number of layers.

**Definition 4.** [10] Let  $\square$  be a special constant symbol. A *context*  $C[\dots]$  is a term of  $T(\Sigma \cup \{\square\}, X)$ . If  $C[\dots]$  is a context with  $n$  occurrences of  $\square$  and  $t_1, \dots, t_n$  are terms, then  $C[t_1, \dots, t_n]$  is the result of replacing from left to right the occurrences of  $\square$  with  $t_1, \dots, t_n$ . Let  $t = C[t_1, \dots, t_n] \in T(\Sigma, X)$  with  $C[\dots] \neq \square$ . We write  $t = C[[t_1, \dots, t_n]]$  if  $C[\dots] \in T(\Sigma_\alpha \cup \{\square\}, X)$  and  $root(t_1), \dots, root(t_n) \in \Sigma_\beta$  for some  $\alpha, \beta \in \{1, 2\}$  with  $\alpha \neq \beta$ . The topmost homogeneous part of  $t$  and the rank of  $t$  are defined by

$$\begin{aligned} top(t) &= \begin{cases} t & \text{if } t \in T(\Sigma_1, X) \cup T(\Sigma_2, X) \\ C[\dots] & \text{if } t = C[[t_1, \dots, t_n]] \end{cases} \\ rank(t) &= \begin{cases} 1 & \text{if } t \in T(\Sigma_1, X) \cup T(\Sigma_2, X) \\ 1 + \max\{rank(t_j) \mid 1 \leq j \leq n\} & \text{if } t = C[[t_1, \dots, t_n]] \end{cases} \end{aligned}$$

**Example 5.** Let  $\Sigma_1 = \{f, a\}$  and  $\Sigma_2 = \{g, b\}$ . Let  $t = f(b, f(b, g(a)))$ . Note that  $f(\square, f(\square, \square))[b, b, g(a)] = t$ . Moreover we have  $f(\square, f(\square, \square)) \in$

$T(\Sigma_1 \cup \{\square\}, X)$ , and  $root(b) = b \in \Sigma_2$ ,  $root(g(a)) = g \in \Sigma_2$ . Therefore we can write  $t = f(\square, f(\square, \square))\llbracket b, b, g(a) \rrbracket$ . Consequently  $top(t) = f(\square, f(\square, \square))$ .  $rank(t) = 1 + \max(rank(b), rank(b), rank(g(a))) = 1 + \max(1, 1, 1 + rank(a)) = 1 + \max(1, 1, 1 + 1) = 1 + 2 = 3$ .

**Remark 6.** If position  $q$  is below  $p$  in  $t$ , i.e.  $q \geq p$ , then  $rank(t|_q) \leq rank(t|_p)$ .

**Definition 7.** Let  $D : t_1 \hookrightarrow t_2 \hookrightarrow \dots$  be a rewrite derivation w.r.t.  $R_1 \cup R_2$ , and consider the set  $Pos(D)$  of rewrite positions in  $D$ .

Let  $posMin(D) = \{u \in Pos(D) \mid \forall p \in Pos(D), \neg(p < u)\}$ . In other words,  $posMin(D)$  is the set of outer positions of  $Pos(D)$ .

**Remark 8.**  $\forall u \in posMin(D), \forall p \in Pos(D), (p \geq u \vee p \parallel u)$ .  
 $\forall i, posMin(D) \subseteq Pos(t_i)$ . Then  $posMin(D)$  is finite (even if  $Pos(D)$  is not).

**Definition** Let  $D : t_1 \hookrightarrow t_2 \hookrightarrow \dots$ . Thanks to Remark 8, we can define : For each  $i$  let  $rank(t_i, D) = \max_{u \in posMin(D)} (rank(t_i|_u))$ , and  $rank(D) = rank(t_1, D)$ . Let  $D|_k : t_k \hookrightarrow t_{k+1} \hookrightarrow \dots$  denote the sub-derivation of  $D$  starting from  $t_k$ .

**Example 9.** Let  $\Sigma_1 = \{f, g\}$ ,  $R_1 = \{\langle f, 2 \rangle : g(x) \rightarrow x\}$ , and  $\Sigma_2 = \{a, b, c, d\}$ ,  $R_2 = \{\langle f, 1 \rangle : a \rightarrow b, \langle f, 1 \rangle : b \rightarrow a, \langle f, 2 \rangle, \langle g, 1 \rangle : c \rightarrow d\}$ .

Consider the infinite derivation  $D$  w.r.t.  $R_1 \cup R_2$  :

$t_1 = f(a, g(c)) \hookrightarrow_{[2,1]} t_2 = f(a, g(d)) \hookrightarrow_{[1]} t_3 = f(b, g(d)) \hookrightarrow_{[2]} t_4 = f(b, d) \hookrightarrow_{[1]} t_5 = f(a, d) \hookrightarrow_{[1]} t_4 \hookrightarrow \dots$

We have  $Pos(D) = \{2.1, 1, 2\}$ , then  $posMin(D) = \{1, 2\}$ .

$rank(D) = rank(t_1, D) = \max(rank(a), rank(g(c))) = \max(1, 2) = 2$ .

$rank(t_1, D) = rank(t_2, D) = rank(t_3, D) = 2$ , and  $rank(t_4, D) = rank(t_5, D) = 1$ .

**Lemma 10** Considering the previous definition :

1.  $\forall i, \forall u \in posMin(D), rank(t_i|_u) \geq rank(t_{i+1}|_u)$ .
2.  $\forall i, rank(t_i, D) \geq rank(t_{i+1}, D)$ , i.e. the rank cannot increase.
3.  $rank(D|_k) \leq rank(D)$  and  $(rank(t_k, D) < rank(D) \Rightarrow rank(D|_k) < rank(D))$ .

PROOF. Items 1, 2 come by Definitions 3, 4, 7 and definition before Example 9. About Item 3, note that  $\forall v \in posMin(D|_k), \exists u \in posMin(D), v \geq u$  because  $Pos(D|_k) \subseteq Pos(D)$ , and consider Remark 6.

**Definition 11.**  $D : t_1 \hookrightarrow t_2 \hookrightarrow \dots$  is *monotonic* if  $\forall i, rank(t_i, D) = rank(t_1, D)$ . An infinite derivation is of *minimal rank* w.r.t.  $R_1 \cup R_2$ , if any derivation w.r.t.  $R_1 \cup R_2$  of smaller rank is finite.

**Lemma 12** Every infinite derivation of minimal rank is monotonic.

PROOF. If  $D$  is not monotonic,  $rank(t_k, D) < rank(t_1, D)$  for some  $k$ . Then  $rank(D|_k) < rank(D)$  and  $D|_k$  is infinite, i.e.  $D$  is not of minimal rank.

Derivation  $D$  of Example 9 is not monotonic, because  $\text{rank}(t_4, D) = 1$  whereas  $\text{rank}(t_1, D) = 2$ . On the other hand,  $\text{posMin}(D|_4) = \{1\}$  and  $\text{rank}(D|_4) = \text{rank}(t_4, D|_4) = \text{rank}(b) = 1$ . Consequently  $D$  is infinite but is not of minimal rank, because  $D|_4$  is also infinite with  $\text{rank}(D|_4) = 1 < 2 = \text{rank}(D)$ .

**Definition 13.** Derivation  $D$  is *singleton* if  $\text{posMin}(D)$  has only one element.

Note that if  $\text{posMin}(D) = \{u\}$ , then  $\forall p \in \text{Pos}(D), p \geq u$ .

**Lemma 14** *If  $D : t_1 \hookrightarrow t_2 \hookrightarrow t_3 \hookrightarrow \dots$  is infinite, there exist  $u \in \text{posMin}(D)$  and an infinite singleton derivation  $D_u^s : t_1 \hookrightarrow \dots$  s.t.  $\text{Pos}(D_u^s) = \{p \in \text{Pos}(D) \mid p \geq u\}$ ,  $\text{posMin}(D_u^s) = \{u\}$ , and  $\text{rank}(D_u^s) \leq \text{rank}(D)$ .*

PROOF. Let  $u \in \text{posMin}(D)$ . Let  $I_u = \{i_1, i_2, \dots\} \subseteq \mathbb{N}$  be such that  $i_1 = 1$  (then  $t_{i_1} = t_1$ ), and  $t_{i_2}, t_{i_3}, \dots$  be the terms of  $D$  obtained by a rewrite step at a position below  $u$ . Thus for each  $k$ ,  $t_{i_k} \xrightarrow*_{[p||u]} \xrightarrow{[p_{i_k} \geq u]} t_{i_{k+1}}$ . Since the rewrite steps at Positions  $p \parallel u$  do not change the prefix of  $p_{i_k} \geq u$ , we get the derivation  $D_u^s : t_{i_1} = t_1 \xrightarrow{[p_{i_1}]} t_1[t_{i_2}|u]_u \xrightarrow{[p_{i_2}]} t_1[t_{i_3}|u]_u \hookrightarrow \dots$ . Note that  $\text{posMin}(D_u^s) = \{u\}$ .

Let  $I = \{1, 2, \dots\}$  be the indices of terms of  $D$ . We have  $\cup_{u \in \text{posMin}(D)} I_u = I$ .  $I$  is infinite since  $D$  is infinite, and by Remark 8  $\text{posMin}(D)$  is finite, therefore there is one (or several)  $u \in \text{posMin}(D)$  such that  $I_u$  is infinite, i.e.  $D_u^s$  is infinite.  $\text{rank}(D_u^s) = \text{rank}(t_1|_u) \leq \max_{v \in \text{posMin}(D)} (\text{rank}(t_1|_v)) = \text{rank}(t_1, D) = \text{rank}(D)$ .

**Example 15.** Consider the infinite derivation  $D$  of Example 9 again.  $D$  is not singleton since  $\text{posMin}(D) = \{1, 2\}$ . With  $u = 1$ , we get the infinite singleton derivation  $D_u^s : t_1 = f(a, g(c)) \xrightarrow{[1]} f(b, g(c)) \xrightarrow{[1]} f(a, g(c)) \hookrightarrow \dots$ . Therefore  $\text{rank}(D_u^s) = \text{rank}(t_1, D_u^s) = \text{rank}(a) = 1 < 2 = \text{rank}(D)$ , which shows again that  $D$  is not of minimal rank. Moreover  $D_u^s$  is infinite, of minimal rank, and monotonic, since by definition, a rank value is strictly positive.

**Definition 16.** Let  $D : t_1 \hookrightarrow t_2 \hookrightarrow \dots$  be a rewrite derivation w.r.t.  $R_1 \cup R_2$ . The step  $t_i \xrightarrow{[p]} t_{i+1}$  is

- *outer* if it applies in  $\text{top}(t_i|_u)$  for  $u \in \text{posMin}(D)$ . Otherwise it is *inner*.
- *destructive at level 1* if  $p \in \text{posMin}(D)$ , and  $\text{root}(t_i|_p) \in \Sigma_\alpha$ ,  $\text{root}(t_{i+1}|_p) \in \Sigma_\beta$  with  $\alpha, \beta \in \{1, 2\}$  and  $\alpha \neq \beta$ .
- *destructive at level 2* if  $\exists u \in \text{posMin}(D)$ ,  $t_i|_u = C[s_1, \dots, s_m, \dots, s_n]$ ,  $t_{i+1}|_u = C[s_1, \dots, s'_m, \dots, s_n]$ ,  $\text{root}(s_m) \in \Sigma_\alpha$ ,  $\text{root}(s'_m) \in \Sigma_\beta$ , with  $\alpha \neq \beta$ .

**Remark 17.** If a rewrite step is destructive, then the rewrite rule is collapsing. A step destructive at level 1 is necessarily outer, and a step destructive at level 2 is necessarily inner.

In the derivation  $D$  of Example 9, the step  $t_1 \hookrightarrow t_2$  is inner whereas all other steps are outer. The step  $t_3 \hookrightarrow t_4$  is destructive at level 1.

**Lemma 18** *Let  $D : t_1 \hookrightarrow t_2 \hookrightarrow \dots$  be a rewrite derivation w.r.t.  $R_1 \cup R_2$ . If  $D$  is singleton with  $\text{posMin}(D) = \{u\}$  and  $t_i \hookrightarrow t_{i+1}$  is destructive at level 1, then  $\text{rank}(t_i|_u) > \text{rank}(t_{i+1}|_u)$ . Therefore  $\text{rank}(t_i, D) > \text{rank}(t_{i+1}, D)$ .*

PROOF. Necessarily  $t_i \hookrightarrow_{[u, L:l \rightarrow x, \sigma]} t_{i+1}$ , with  $x \in \text{Var}(l)$  because of Remark 17, and  $t_{i+1}|_u = \sigma(x)$ . Moreover  $\text{root}(t_i|_u) \in \Sigma_\alpha$ ,  $\text{root}(\sigma(x)) \in \Sigma_\beta$  with  $\alpha \neq \beta$ . Then  $t_i|_u = C[\dots, \sigma(x), \dots]$ , and  $\text{rank}(t_i|_u) \geq 1 + \text{rank}(\sigma(x)) = 1 + \text{rank}(t_{i+1}|_u)$ .

**Lemma 19** *Every singleton infinite derivation of minimal rank, contains infinitely many outer steps.*

PROOF. Let  $D : t_1 \hookrightarrow t_2 \hookrightarrow \dots$  with  $\text{posMin}(D) = \{u\}$ . Assume that  $D$  contains finitely many outer steps. Let  $k$  be the least index such that no outer step appears in  $D$  after  $t_k$ . Let  $D|_k$  be the sub-derivation of  $D$  starting from  $t_k$ . Note that all steps of  $D|_k$  are inner in  $D$ . Then  $t_k|_u = C[\dots]$  and rewrite positions of  $D|_k$  occur below  $C$ . Therefore if  $p \in \text{posMin}(D|_k)$ , then  $p$  occurs below  $C$  in  $t_k$ . Thus  $\text{rank}(D) = \text{rank}(t_1, D) \geq \text{rank}(t_k, D) = \text{rank}(t_k|_u) \geq 1 + \text{rank}(t_k|_p) > \text{rank}(t_k|_p)$ . Therefore  $\text{rank}(D) > \max_{p \in \text{posMin}(D|_k)}(\text{rank}(t_k|_p)) = \text{rank}(D|_k)$ . So  $D|_k$  is infinite and its rank is smaller than that of  $D$ , which is impossible since  $D$  is of minimal rank.

The singleton infinite derivation  $D_u^s$  of Example 15 is of minimal rank, monotonic, and all steps are outer. This is not a contradiction with the following proposition since  $R_2$  of Example 15 is not terminating.

**Proposition 20** *If  $R_1 \cup R_2$  is not terminating whereas  $R_1$  and  $R_2$  are terminating, there exists an infinite derivation  $D_u^s$  of minimal rank, which is singleton and monotonic, and which contains infinitely many outer steps. Moreover,  $D_u^s$  contains infinitely many inner steps destructive at level 2.*

PROOF. There is an infinite derivation  $D$  of minimal rank w.r.t.  $R_1 \cup R_2$ . By Lemma 14 there is an infinite singleton derivation  $D_u^s$  with  $\text{posMin}(D_u^s) = \{u\}$  and  $\text{rank}(D_u^s) \leq \text{rank}(D)$ . Since  $D$  is of minimal rank,  $\text{rank}(D_u^s) = \text{rank}(D)$  and  $D_u^s$  is of minimal rank.  $D_u^s$  is monotonic by Lemma 12. And by Lemma 19,  $D_u^s$  contains infinitely many outer steps.

Let  $D_u^s : t_1 \hookrightarrow t_2 \hookrightarrow \dots$ . Assume that there are finitely many inner steps destructive at level 2. Let  $k$  be the least index such that no inner step destructive at level 2 appears in  $D_u^s$  after  $t_k$ . Assume  $\text{root}(t_k|_u) \in \Sigma_\alpha$  with  $\alpha \in \{1, 2\}$ . Let  $t_j \hookrightarrow_{[p]} t_{j+1}$  be a step of  $D_u^s$  with  $j \geq k$ .  $D_u^s$  is monotonic, and by Lemma 18, the step is not destructive at level 1. If the step is outer, it is not destructive, then  $t_j[\text{top}(t_j|_u)]_u \hookrightarrow_{[p]} t_{j+1}[\text{top}(t_{j+1}|_u)]_u$  by a rule of  $R_x$ . If the step is inner, it is not destructive at level 2, then  $\text{top}(t_j|_u) = \text{top}(t_{j+1}|_u)$ . Hence,  $D_u^s$  has infinitely many outer steps, and  $R_\alpha$  is not terminating.

**Definition 21.**  $v \in \text{Dir}(\Sigma)^*$  is *active* if  $v \in L'$  for some  $L' : l' \rightarrow r' \in R_1 \cup R_2$ .

Intuitively, “duplicating” means that the rule can duplicate active sub-terms.



**Definition 22.** Let  $s$  be a term and  $v, w \in \text{Dir}(\Sigma)^*$ .

- We define the multiset of active variables of  $s$  instantiated by  $w$ , under  $v$ ,  $\text{actVar}(s, v, w) = \{x \mid \exists p \in \text{posVar}(s), s(p) = x \wedge v.\text{path}(s, p).w \text{ is active}\}_{\text{mult}}$
- The rule  $L : l \rightarrow r \in R_1 \cup R_2$  is *non-duplicating* if :  
 $\forall v \in L, \forall w \in \text{Dir}(\Sigma)^*, \text{actVar}(r, v, w) \subseteq \text{actVar}(l, v, w)$ .
- A rewrite step using a duplicating rule, is said duplicating.

**Example 23.** Consider  $\Sigma_1 = \{f, g, h\}$ ,  $\Sigma_2 = \{a, b\}$ , and  $R_1 \cup R_2$  with

$$R_1 = \{\epsilon : g(x) \rightarrow h(x), \epsilon : g(x) \rightarrow f(x, x)\},$$

$$R_2 = \{\langle f, 1 \rangle \mid \langle g, 1 \rangle \mid \langle h, 1 \rangle.\langle h, 1 \rangle : a \rightarrow b\}, \text{ where '}' \text{ denotes the union.}$$

The first rule of  $R_1$  is duplicating because  $\text{actVar}(h(x), \epsilon, \langle h, 1 \rangle) = \{x\}_{\text{mult}} \not\subseteq \text{actVar}(g(x), \epsilon, \langle h, 1 \rangle) = \emptyset$ . The second rule of  $R_1$  is not duplicating because  $\text{actVar}(f(x, x), \epsilon, \epsilon) = \{x\}_{\text{mult}}$  is included into  $\text{actVar}(g(x), \epsilon, \epsilon) = \{x\}_{\text{mult}}$ , and if  $w \neq \epsilon$ ,  $\text{actVar}(f(x, x), \epsilon, w) = \text{actVar}(g(x), \epsilon, w) = \emptyset$ .

Note that the linear rule  $g(x) \rightarrow h(x)$  is also duplicating in the context-sensitive framework if the replacement map  $\mu$  defines  $\mu(h) = \{1\}$  and  $\mu(g) = \emptyset$ .

This is decidable since  $L$  is regular. For any term  $s$ , if  $s(p) = x$  the language of prefixes and instances (separated by  $\&$ )  $L_{s,p} = \{v.\&.w \in (\text{Dir}(\Sigma) \cup \{\&\})^* \mid v \in L \wedge v.\text{path}(s, p).w \text{ is active}\}$  is recognized by a finite automaton (see below).

If the rule is left-linear, for each variable  $x$  occurring at positions  $q_1, \dots, q_n$  in  $r$ , and at position  $p$  in  $l$ , we check that  $L_{r,q_i} \subseteq L_{l,p}$  for all  $i$ , and  $L_{r,q_i} \cap L_{r,q_j} = \emptyset$  for all  $i, j$  with  $i \neq j$ .

Otherwise, for each variable  $x$  occurring at positions  $q_1, \dots, q_n$  in  $r$ , and at positions  $p_1, \dots, p_m$  in  $l$ , and for each non-empty subset  $q_{i_1}, \dots, q_{i_k}$  of  $q_1, \dots, q_n$ , we check that  $L_{r,q_{i_1}} \cap \dots \cap L_{r,q_{i_k}} \subseteq \cup_{\{j_1, \dots, j_k\} \subseteq \{1, \dots, m\}} (L_{l,p_{j_1}} \cap \dots \cap L_{l,p_{j_k}})$ . Note that each subset  $\{j_1, \dots, j_k\}$  should have  $k$  elements. If  $k > m$ , it is impossible, which means that  $\cup_{\{j_1, \dots, j_k\} \subseteq \{1, \dots, m\}} (L_{l,p_{j_1}} \cap \dots \cap L_{l,p_{j_k}}) = \emptyset$ .

To compute  $L_{s,p}$ , we first compute an automaton  $\mathcal{A}_{L_\cap}$  that recognizes  $L_\cap = (L.\{\text{path}(s, p)\}.\text{Dir}(\Sigma)^*) \cap (\cup_{L'.l \rightarrow r' \in R_1 \cup R_2} L') = \{v.\text{path}(s, p).w \mid v.\&.w \in L_{s,p}\}$ , using the usual techniques to get concatenation, union, intersection of regular languages. Assume that the initial automata  $\mathcal{A}_L, \mathcal{A}_{\{\text{path}(s, p)\}}, \mathcal{A}_{\text{Dir}(\Sigma)^*}, \mathcal{A}_{\cup L'}$  have disjoint states. Because of the intersection, the states of  $\mathcal{A}_{L_\cap}$  are pairs of states of the initial automata. To get  $\mathcal{A}_{L_{s,p}}$  from  $\mathcal{A}_{L_\cap}$ , we remove all states (and the corresponding transitions) of the form  $(q_1, q_2)$  s.t.  $q_1$  is a state of  $\mathcal{A}_{\{\text{path}(s, p)\}}$ , and we add all transitions of the form  $((q_f, q_3), \&, (q_i, q_4))$  s.t.  $q_f$  is a final state of  $\mathcal{A}_L$ , and  $q_i$  is an initial state of  $\mathcal{A}_{\text{Dir}(\Sigma)^*}$ , and  $q_3, q_4$  are states of  $\mathcal{A}_{\cup L'}$ .

**Definition 24.** Let  $D_u^s$  be a singleton derivation with  $\text{posMin}(D_u^s) = \{u\}$ , and  $s_j$  be a term in  $D_u^s$ . Let us write  $s_j|_u = C[[t_1, \dots, t_n]]$ , and for each  $i$ , let  $p_i \in \text{Pos}(s_j)$  be the position of  $t_i$  in  $s_j$ .

We define the multiset  $\#s_j = \{\text{rank}(t_i) \mid \text{path}(s_j, p_i) \text{ is active}\}_{\text{mult}}$ .

$\geq^{\text{mult}}$  is the multiset extension of the usual ordering  $\geq$  over positive integers, i.e. it is the reflexive-transitive closure of  $\succ$  where  $M + \{m\} \succ M + \{n_1, \dots, n_k\}$  if  $\forall i \in [1..k], m > n_i$ . Symbol '+' adds occurrences to the multiset  $M$ .



**Lemma 25** Consider any rewrite step  $s_j \xrightarrow{[p]} s_{j+1}$  in  $D_u^s$ . If the step is :

1. outer, non-destructive and non-duplicating, then  $\#s_j \geq^{mult} \#s_{j+1}$ .
2. inner and non-destructive, then  $\#s_j \geq^{mult} \#s_{j+1}$ .
3. inner and destructive, then  $\#s_j >^{mult} \#s_{j+1}$ .

PROOF. Item 1. Outer and non-destructive implies  $s_j[C]_u \xrightarrow{[p, L: l \rightarrow r]} s_{j+1}[C']_u$  with  $C = \text{top}(s_j|_u)$  and  $C' = \text{top}(s_{j+1}|_u)$ . Let  $x \in \text{Var}(l)$  and  $e$  be a position. Let  $P_x \subseteq \{p_1, \dots, p_n\}$  be the biggest set s.t. for all  $p_x \in P_x$ ,  $\text{path}(s_j, p_x)$  is active, and  $p_x = p.q_x.e$  with  $l(q_x) = x$ . Note that  $\text{path}(s_j, p_x) = v.\text{path}(l, q_x).w$  with  $v = \text{path}(s_j, p) \in L$ , and  $w = \text{path}(s_j|_{p.q_x}, e)$  does not depend on the choice of  $q_x$ , and the choice of  $p_x$ . Therefore  $\{l(q_x) \mid \exists p_x \in P_x, p_x = p.q_x.e\}_{mult} \cap \{x, \dots\}_{mult} = \text{actVar}(l, v, w)$ .

Let  $Q'_x = \{q'_x \mid r(q'_x) = x \wedge v.\text{path}(r, q'_x).w \text{ is active}\}$ . Then  $\{r(q'_x) \mid \exists q'_x \in Q'_x\}_{mult} \cap \{x, \dots\}_{mult} = \text{actVar}(r, v, w)$ . Since the rule is not duplicating and  $v \in L$ ,  $\text{actVar}(l, v, w) \supseteq \text{actVar}(r, v, w)$ . So the cardinal  $|P_x| \geq |Q'_x| = |p.Q'_x.e|$ .

Thus, among  $t_1, \dots, t_n$  in  $s_j$ , the active elements that are instances of  $x$  are not duplicated at active positions of  $s_{j+1}$ . It holds for all  $x \in \text{Var}(l)$  and for all positions  $e$ . Consequently  $\#s_j \geq^{mult} \#s_{j+1}$ .

Item 2.  $s_j = s_j[C[[t_1, \dots, t_k, \dots, t_n]]]_u \xrightarrow{[p]} s_{j+1} = s_j[C[[t_1, \dots, t'_k, \dots, t_n]]]_u$ . Then  $p \geq p_k$  and  $\text{rank}(t_k) \geq \text{rank}(t'_k)$ . The position of  $t'_k$  in  $s_{j+1}$  is also  $p_k$ , and  $\text{path}(s_{j+1}, p_k) = \text{path}(s_j, p_k)$ . Therefore if  $\text{path}(s_{j+1}, p_k)$  is active, then  $\text{path}(s_j, p_k)$  also is. Then  $\#s_j \geq^{mult} \#s_{j+1}$ .

Item 3.  $s_j = s_j[C[[t_1, \dots, t_k, \dots, t_n]]]_u \xrightarrow{[p]} s_{j+1} = s_j[C[[t_1, \dots, t'_k, \dots, t_n]]]_u$ . Then  $p = p_k$  and  $\text{rank}(t_k) > \text{rank}(t'_k)$ , and  $p_k$  is active. Let us write  $t'_k = C'[[\alpha_1, \dots, \alpha_m]]$ . Then  $s_{j+1} = s_j[C''[[t_1, \dots, \alpha_1, \dots, \alpha_m, \dots, t_n]]]_u$  and for all  $i$ ,  $\text{rank}(t'_k) > \text{rank}(\alpha_i)$ . Therefore  $\text{rank}(t_k) \in \#s_j$  is replaced by a sub-multiset of  $\{\text{rank}(\alpha_1), \dots, \text{rank}(\alpha_m)\}_{mult}$  to obtain  $\#s_{j+1}$ . Then  $\#s_j >^{mult} \#s_{j+1}$ .

**Proposition 26**  $D_u^s$  of Proposition 20 has at least one outer duplicating step.

PROOF. By contradiction. If all outer steps of  $D_u^s$  are non-duplicating, they are also non-destructive since  $D_u^s$  is monotonic. By Proposition 20,  $D_u^s$  has infinitely many inner steps destructive at level 2. By Lemma 25, there is an infinite decreasing sequence for  $>^{mult}$ , which is impossible ( $>^{mult}$  is well-founded).

**Theorem 27** Termination is modular for totally disjoint pCTRSs.

PROOF. In this case a derivation has no inner step. If termination is not modular, Derivation  $D_u^s$  of Proposition 20 would have infinitely many inner steps.

**Theorem 28** Let  $R_1, R_2$  be two weakly disjoint terminating pCTRSs such that their union  $R_1 \cup R_2$  is not terminating. Then  $R_\alpha$  is duplicating and  $R_\beta$  is collapsing for some  $\alpha, \beta \in \{1, 2\}$  with  $\alpha \neq \beta$ .

PROOF. Derivation  $D_u^s : t_1 \xrightarrow{[p_1]} t_2 \xrightarrow{\dots}$  of Proposition 20 is singleton ( $\text{posMin}(D_u^s) = \{u\}$ ) and monotonic. By Lemma 18,  $D_u^s$  does not have steps

destructive at level 1, then there is  $\alpha \in \{1, 2\}$  s.t.  $\forall i, \text{root}(t_i|_u) \in \Sigma_\alpha$ . Therefore  $\forall i, \text{top}(t_i|_u) \in T(\Sigma_\alpha \cup \{\square\}, X)$ . So if any step  $t_i \hookrightarrow_{[p_i]} t_{i+1}$  is outer, then the rewrite rule is in  $R_\alpha$ . By Proposition 26,  $R_\alpha$  has at least one duplicating rule.

By Proposition 20,  $D_u^s$  has a step  $t_j \hookrightarrow_{[p_j]} t_{j+1}$  destructive at level 2. By Definition 16,  $t_j|_u = C[s_1, \dots, s_m, \dots, s_n]$ , and  $\text{root}(s_m) \in \Sigma_\beta$  with  $\beta \neq \alpha$  since  $C = \text{top}(t_j|_u) \in T(\Sigma_\alpha \cup \{\square\}, X)$ . This step applies to the root of  $s_m$ , then with a rule of  $R_\beta$ . By Remark 17 this rule is collapsing, then  $R_\beta$  is collapsing.

## References

- [1] N. Andrianarivelo, V. Pelletier, and P. Réty. Transforming prefix-constrained or controlled rewrite systems. In *Symposium on Symbolic in Software Science (SCSS)*, volume 45 of *EPiC Series in Computing*, 2017.
- [2] M. Fernández and J.-P. Jouannaud. Modular termination of term rewriting systems revisited. In *5th COMPASS Workshop*, volume 906 of *LNCS*, 1994.
- [3] Bernhard Gramlich. Generalized sufficient conditions for modular termination of rewriting. *Applicable Algebra in Engineering, Communication and Computing*, 5:131–158, 1994.
- [4] Bernhard Gramlich. Modularity in term rewriting revisited. *Theoretical Computer Science*, 464:3–19, 2012.
- [5] Bernhard Gramlich and Salvador Lucas. Modular termination of context-sensitive rewriting. In *SIGPLAN conference on Principles and practice of declarative programming*, pages 50–61. ACM, 2002.
- [6] F. Jacquemard, Y. Kojima, and M. Sakai. Term Rewriting with Prefix Context Constraints and Bottom-Up Strategies. In *Conference on Automated Deduction (CADE)*, volume 9195 of *LNCS*, pages 137–151, 2015.
- [7] S. Lucas. Context-sensitive rewriting. *ACM Comput. Surv.*, 53(4), 2020.
- [8] Aart Middeldorp. A sufficient condition for the termination of the direct sum of term rewriting systems. In *Symposium on Logic in Computer Science (LICS)*, pages 396–401. IEEE Computer Society, 1989.
- [9] Enno Ohlebusch. On the modularity of termination of term rewriting systems. *Theoretical Computer Science*, 136(2):333–360, 1994.
- [10] Enno Ohlebusch. *Advanced Topics in Term Rewriting*. Springer, 2002.
- [11] Michaël Rusinowitch. On termination of the direct sum of term-rewriting systems. *Information Processing Letters*, 26(2):65–70, 1987.
- [12] Y. Toyama, J.-W. Klop, and H.-P. Barendregt. Termination for the direct sum of left-linear term rewriting systems. In *Conference on Rewriting Techniques and Applications (RTA)*, volume 355 of *LNCS*, 1989.
- [13] Yoshihito Toyama. Counterexamples to termination for the direct sum of term rewrite systems. *Information Processing Letters*, 25(3):141–143, 1987.

### **Declaration of interests**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Nirina Andrianarivelo  
Pierre Réty