# On the Confluence of Linear Shallow Term Rewrite Systems

Guillem Godoy*, Ashish Tiwari**, and Rakesh Verma***

Technical University of Catalonia
Jordi Girona 1, Barcelona, Spain
*ggodoy@lsi.upc.es*

SRI International
Menlo Park, CA
*tiwari@csl.sri.com*

Computer Science Dept
Univ of Houston, TX
*rmverma@cs.uh.edu*

**Abstract.** We show that the confluence of shallow linear term rewrite systems is decidable. The decision procedure is a nontrivial generalization of the polynomial time algorithms for deciding confluence of ground and restricted non-ground term rewrite systems presented in [13, 2]. Our algorithm has a polynomial time complexity if the maximum arity of a function symbol in the signature is considered a constant. We also give EXPTIME-hardness proofs for reachability and confluence of shallow term rewrite systems.

## 1 Introduction

Programming language interpreters, proving equations (e.g. $x^3 = x$ implies the ring is Abelian), abstract data types, program transformation and optimization, and even computation itself (e.g., Turing machine) can all be specified by a set of rules, called a rewrite system. The rules are used to replace ("reduce") subexpressions of given expressions by other expressions (usually equivalent ones in some sense). A fundamental property of a rewrite system is the confluence or Church-Rosser property. Informally, confluence states that if an expression $a$ can be reduced (in zero or more steps) to two different expressions $b$ and $c$, then there is a common expression $d$ to which $b$ and $c$ can be reduced in zero or more steps. Confluence implies uniqueness of normal ("irreducible") forms and helps to "determinise" their search by avoiding backtracking.

In general, confluence is well-known to be undecidable; however, it is known to be decidable for terminating systems [8] and for the subclass of arbitrary variable-free ("ground") systems [4, 11]. Ground systems include as a subclass the tree automata model, which has important computer science applications. The previous decidability proofs of confluence for ground systems [4, 11] were based on tree-automata techniques and showed that this problem was in EXPTIME, but no nontrivial lower bounds were known. Hence the exact complexity of this problem was open until last year, when a series of papers [7, 2, 13, 6] culminated in a polynomial time algorithm for this problem for shallow and rule-linear systems, which include ground systems as a special case. In a shallow system variables in the rules cannot appear at depth more than one. Shallow

* Partially supported by the Spanish CICYT project MAVERISH ref. TIC2001-2476-C03-01.
** Research supported in part by DARPA under the MoBIES and SEC programs administered by AFRL under contracts F33615-00-C-1700 and F33615-00-C-3043, and NSF CCR-0082560.
*** Research supported in part by NSF grant CCR-9732186.

systems have been well-studied in other contexts [10, 3]. Linearity in [13, 6], called rule-linearity here, means each variable can appear at most once in the entire rule. Thus, commutativity ($x+y = y+x$) is a shallow equation but *not* rule-linear, and associativity is neither shallow nor rule-linear.

In this paper, we establish decidability of confluence for shallow systems in which the left-hand side and right-hand side are *independently* linear, i.e., a variable can have two occurrences in a rule—once in each side of the rule. In fact, the decision procedure runs in polynomial time if we assume that the maximum arity of a function symbol in the signature is a constant. The results in this paper subsume the shallow rule-linear systems of [13] as a special case. The algorithm is a nontrivial generalization of the algorithms in [2, 13]. We introduce a notion of marked terms and marked rewriting, and then generalize the central concept of top stability in [2]. The conditions to be checked by the algorithm are also generalized and the constructions are more involved. We also prove that the reachability, joinability and confluence problems are all EXPTIME-hard for shallow non-linear systems and all are known to be undecidable for linear non-shallow systems [15], which indicates that the linearity and shallowness assumptions are fairly tight.

### 1.1 Preliminaries

Let $\mathcal{F}$ be a (finite) set of function symbols with an arity function $arity\colon \mathcal{F} \to I\!N$. Function symbols $f$ with $arity(f) = n$, denoted by $f^{(n)}$, are called *$n$-ary* symbols (when $n = 1$, one says *unary* and when $n = 2$, *binary*). If $arity(f) = 0$, then $f$ is a *constant symbol*. Let $\mathcal{X}$ be a set of variable symbols. The set of terms over $\mathcal{F}$ and $\mathcal{X}$, denoted by $\mathcal{T}(\mathcal{F}, \mathcal{X})$, is the smallest set containing all constant and variable symbols such that $f(t_1, \ldots, t_n)$ is in $\mathcal{T}(\mathcal{F}, \mathcal{X})$ whenever $f \in \mathcal{F}$, $arity(f) = n$, and $t_1, \ldots, t_n \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. A *position* is a sequence of positive integers. If $p$ is a position and $t$ is a term, then by $t|_p$ we denote the *subterm of $t$ at position $p$*: we have $t|_\lambda = t$ (where $\lambda$ denotes the empty sequence) and $f(t_1, \ldots, t_n)|_{i.p} = t_i|_p$ if $1 \leq i \leq n$ (and is undefined if $i > n$). We also write $t[s]_p$ to denote the term obtained by replacing in $t$ the subterm at position $p$ by the term $s$. For example, if $t$ is $f(a, g(b, h(c)), d)$, then $t|_{2.2.1} = c$, and $t[d]_{2.2} = f(a, g(b, d), d)$. By $|s|$ we denote the *size* (number of symbols) of a term $s$: we have $|a| = 1$ if $a$ is a constant symbol or a variable, and $|f(t_1, \ldots, t_n)| = 1 + |t_1| + \ldots + |t_n|$. The *depth* of a term $s$ is 0 if $s$ is a variable or a constant, and $1 + max_i\, depth(s_i)$ if $s = f(s_1, \ldots, s_m)$. Terms with depth 0 are denoted by $\alpha, \beta$, with possible subscripts.

If $\to$ is a binary relation on a set $S$, then $\to^+$ is its transitive closure, $\leftarrow$ is its inverse, and $\to^*$ is its reflexive-transitive closure. Two elements $s$ and $t$ of $S$ are called *joinable* by $\to$, denoted $s \downarrow t$, if there exists a $u$ in $S$ such that $s \to^* u$ and $t \to^* u$. The relation $\to$ is called *confluent* or *Church-Rosser* if the relation $\leftarrow^* \circ \to^*$ is contained in $\to^* \circ \leftarrow^*$, that is, for all $s$, $t_1$ and $t_2$ in $S$, if $s \to^* t_1$ and $s \to^* t_2$, then $t_1 \downarrow t_2$. An equivalent definition of confluence of $\to$ is that $\leftrightarrow^*$ is contained in $\to^* \circ \leftarrow^*$, that is, all $s$ and $t$ in $S$ such that $s \leftrightarrow^* t$ are joinable.

A *substitution* $\sigma$ is a mapping from variables to terms. It can be homomorphically extended to a function from terms to terms: using a postfix notation, $t\sigma$ denotes the result of simultaneously replacing in $t$ every $x \in Dom(\sigma)$ by $x\sigma$. Substitutions are sometimes written as finite sets of pairs $\{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$, where each $x_i$ is a variable and each $t_i$ is a term. For example, if $\sigma$ is $\{x \mapsto f(b, y), y \mapsto a\}$, then $g(x, y)\sigma$ is $g(f(b, y), a)$.

A *rewrite rule* is a pair of terms $(l, r)$, denoted by $l \to r$, with left-hand side (lhs) $l$ and right-hand side (rhs) $r$. A *term rewrite system* (TRS) $R$ is a finite set of rewrite rules. We say that $s$ rewrites to $t$ in one step at position $p$ (by $R$), denoted by $s \to_{R,p} t$, if $s|_p = l\sigma$ and $t = s[r\sigma]_p$, for some $l \to r \in R$ and substitution $\sigma$. If $p = \lambda$, then the rewrite step is said to be applied *at the topmost position* (at the root) and is denoted by $s \to_R^r t$; it is denoted by $s \to_R^{nr} t$ otherwise. The rewrite relation $\to_R$ induced by $R$ on $\mathcal{T}(\mathcal{F}, \mathcal{X})$ is defined by $s \to_R t$ if $s \to_{R,p} t$ for some position $p$.

A *(rewrite) derivation or proof* (from $s$) is a sequence of rewrite steps (starting from $s$), that is, a sequence $s \to_R s_1 \to_R s_2 \to_R \ldots$. The *size* $|R|$ of a TRS $R$ of the form $\{l_1 \to r_1, \ldots, l_n \to r_n\}$ is $|l_1| + |r_1| + \ldots + |l_n| + |r_n|$.

**Definition 1.** *A term $t$ is called*

- linear *if no variable occurs more than once in $t$.*
- shallow *if no variable occurs in $t$ at depth greater than 1, i.e., if $t|_p$ is a variable, then $p$ is a position of length zero or one.*
- flat *if $t$ is a non-constant term of the form $f(s_1, \ldots, s_n)$ where all $s_i$ are variables or constants.*

**Definition 2.** *Let $R$ be a TRS.*
*A term $s$ is* reachable *from $t$ by $R$ if $t \to_R^* s$.*
*Two terms $s$ and $t$ are* equivalent *by $R$ if $s \leftrightarrow_R^* t$.*
*Two terms $s$ and $t$ are* joinable *by $R$, denoted by $s \downarrow_R t$, if they are joinable by $\to_R$.*
*A term $s$ is $R$-*irreducible *if there is no term $t$ s.t. $s \to_R t$.*
*The TRS $R$ is* confluent *if the relation $\to_R$ is confluent on $\mathcal{T}(\mathcal{F}, \mathcal{X})$.*

We assume that $R$ is a shallow and linear term rewrite system, that is, if $s \to t$ is a rule in $R$, then $s$ and $t$ are both linear and shallow terms. Unlike previous results in [13, 6], the terms $s$ and $t$ are allowed to share variables.

## 2 Confluence of Shallow and Linear Rewrite Systems

Assuming that the maximum arity of a function symbol in $\mathcal{F}$ is bounded by a constant, we show that confluence of shallow and linear term rewrite system $R$ over $\mathcal{F}$ can be decided in polynomial time. The proof of this fact uses suitable generalizations of the techniques in [2, 13]. In Section 2.1 we argue that without loss of generality, we can restrict the signature $\mathcal{F}$ to contain exactly one function symbol with nonzero arity. Thereafter, we transform the rewrite system $R$ into a flat rewrite system in Section 2.2. The flat linear term rewrite system is saturated under ordered chaining inference rule in Section 2.3 to construct a rewrite closure, which has several useful properties. The rest of the proof relies on the notion of top-stable and marked top-stable terms (Section 2.4), the ability to compute these sets (Section 2.5), and relating confluence of a saturated flat linear rewrite system to efficiently checkable properties over these sets (Section 2.6).

### 2.1 Simplifying the Signature

Terms over an arbitrary signature $\mathcal{F}$ can be encoded by terms over a signature $\mathcal{F}'$ containing at most one function symbol with non-zero arity. We may assume that $\mathcal{F}$ contains at least one constant $e$ that does not appear in $R$.

**Proposition 1.** *There exists an injective mapping $\sigma$ from terms over an arbitrary $\mathcal{F}$ to terms over a signature $\mathcal{F}'$ containing exactly one function symbol (with non-zero fixed arity) such that if $R'$ is defined as $\{\sigma(s) \to \sigma(t) : s \to t \in R\}$, then $R$ is confluent if, and only if, $R'$ is confluent.*

*Proof.* (Sketch) Let $m$ be one plus the maximum arity of any function symbol in $\mathcal{F}$. Define the new signature $\mathcal{F}'$ as

$$\mathcal{F}' = \{h^{(0)} : h^{(l)} \in \mathcal{F}, l > 0\} \cup \{f^{(m)}\} \cup \{c^{(0)} : c^{(0)} \in \mathcal{F}\},$$

where $f$ is a new symbol. Define the map $\sigma$ as follows: for each $h \in \mathcal{F}$ with arity $l > 0$,

$$\sigma(h(t_1, \ldots, t_l)) = f(\sigma(t_1), \ldots, \sigma(t_l), e, \ldots, e, h)$$

where the number of $e$'s above equals $m - l - 1$, and for each $c \in \mathcal{F}$ with arity 0, $\sigma(c) = c$. The mapping $\sigma$ is clearly injective, but not surjective. We can classify terms over $\mathcal{F}'$ into type 1 and type 2 terms (using a simple sorted signature) so that terms of type 1 exactly correspond to $\mathit{Range}(\sigma)$. It is easy to see that there is a bijective correspondence between proofs in $R$ and proofs in $R'$ over terms in $\mathit{Range}(\sigma)$. Combining this observation with a result in [16], which states that proving confluence for arbitrary terms over the signature is equivalent to proving confluence of the well-typed terms according to any many-sorted discipline which is compatible with the rewrite system under consideration, it follows that $R$ is confluent iff $R'$ is confluent.

### 2.2 Flat Representation

In the transformation described in Section 2.1, the properties of being linear and shallow are preserved. We next flatten the term rewrite system so that the depth of each term is at most one. In particular, given a linear shallow term rewrite system $R$, it can be transformed so that each rule in $R$ is of the form

| | | | |
|---|---|---|---|
| $f(\alpha_1, \ldots, \alpha_m) \to c$ | $(F_c)$ | $c \to f(\alpha_1, \ldots, \alpha_m)$ | $(B_c)$ |
| $f(\alpha_1, \ldots, \alpha_m) \to x$ | $(F_x)$ | $x \to f(\alpha_1, \ldots, \alpha_m)$ | $(B_x)$ |
| $f(\alpha_1, \ldots, \alpha_m) \to f(\beta_1, \ldots, \beta_m)$ | $(P_f)$ | $\alpha \to \beta$ | $(P_c)$ |

where each $\alpha_i, \beta_i, \alpha, \beta$ is a depth 0 term (i.e., either a variable or a constant). Rules of the form $F_c$ and $F_x$ are called *forward* rules and denoted by $F$, rules of the form $B_c$ and $B_x$ are called *backward* rules and denoted by $B$, and rules of the form $P_f$ and $P_c$ are called *permutation* rules and denoted by $P$. Rules of the form $B_x$ are called *insertion* rules. We call such a rewrite system $R$ a *flat linear* rewrite system.

This transformation is easily done by replacing each non-constant ground term, say $s$, in $R$ by a new constant, say $c$, and adding a rule $s \to c$ or $c \to s$, depending on whether $s$ occurred on the left- or right-hand side of $R$.

$$\text{Flatten:} \qquad \frac{u[s] \to t}{u[c] \to t, s \to c} \qquad \frac{t \to u[s]}{t \to u[c], c \to s}$$

where $s$ is a non-constant ground term and $c$ is a new constant.

Exhaustive application of these two rules results in a flat linear shallow rewrite system. This transformation can be done in polynomial time, as the number of applications of the above two rules is bounded by the size of the initial rewrite system $R$. It is easily seen to preserve confluence, see [2, 13] for instance.

### 2.3 Rewrite Closure

Let $\succ$ order terms based on their size, that is, $s \succ t$ iff $|s| > |t|$. An application of an $F$-rule results in a smaller term, whereas application of a $B$-rule gives a bigger term in this ordering.

**Definition 3.** *A term $s$ is size-irreducible by $R$ if there exists no term $t$ such that $s \to_R^* t$ and $s \succ t$.*

**Definition 4.** *A derivation $s \to_R^* t$ is said to be increasing if for all decompositions $s \to_R^* s' \to_{l \to r, p} t' \to_R^* t$, there is no step at a prefix position of $p$ in $t' \to_R^* t$.*

Observe that increasing derivations either have no rewrite step at position $\lambda$, or only one at the beginning of the derivation. For simplicity, we eliminate the former case by assuming a dummy rewrite rule $x \to x$ to be in $R$, which can always be applied at the $\lambda$ position in case there is no top step.

A flat linear rewrite system can be saturated under the following ordered chaining inference to give an enlarged flat linear rewrite system with some nice properties.

$$\text{Ordered Chaining:} \qquad \frac{s \to t \qquad w[u] \to v}{w[s]\sigma \to v\sigma} \qquad \frac{s \to w[t] \qquad u \to v}{s\sigma \to w[v]\sigma}$$

where $\sigma$ is the most general unifier of $t$ and $u$, neither $u$ nor $t$ is a variable, and $s \not\succ t$ in the first case and $v \not\succ u$ in the second. Note that these restrictions ensure that ordered chaining preserves flatness and shallowness.

Application of ordered chaining preserves confluence. Moreover, if the maximum arity $m$ is a constant, then saturation under ordered chaining can be performed in polynomial time.

**Lemma 1.** *Let $R = F \cup B \cup P$ be a flat linear rewrite system saturated under the ordered chaining inference rules. If $s \to_R^* t$, then there is a proof of the form $s \to_F^* \circ \to_P^* \circ \to_B^* t$.*

Lemma 1 can be easily established using proof simplification arguments [1]. Similar proofs have been presented before, but for the special case of ground systems [12] and rule-linear shallow rewrite systems [14]. The generalization to linear shallow case is straightforward and the details are skipped here. The process of saturation, in this context, can be interpreted as asymmetric completion [9].

**Lemma 2.** *Let $R$ be a flat linear rewrite system saturated under the ordered chaining inference rule above. If $s$ is size-irreducible (or, equivalently $F$-irreducible) and $s \to_R^* t$, then there is an increasing derivation $s \to_R^* t$.*

*Example 1.* If $R = \{x + y \to y + x, x \to 0 + x\}$, then the chaining inferences add a new rule $x \to x + 0$ to $R$. An increasing derivation for $0 + x \to^* (x + 0) + 0$ is $0 + x \to x + 0 \to (x + 0) + 0$.

### 2.4 Top-Stable Terms, Marked Terms, and Marked Rewriting

In the rest of the paper we assume that $R$ is a flat linear term rewrite system, which is also saturated under the chaining inference rule.

**Definition 5.** *A term $t$ with depth greater than $0$ is said to be* top-stable *if it cannot be reduced to a depth $0$ term. A depth $0$ term $\alpha$ is* top-stabilizable *if it is equivalent to a top-stable term.*

The following is a simple consequence of Lemma 1.

**Lemma 3.** *The set $S_0 = \{f\alpha_1 \ldots \alpha_m : f\alpha_1 \ldots \alpha_m$ is $F$-irreducible$\}$ is the set of all top-stable flat terms.*

The confluence test relies heavily on the concept of top-stable terms and depth $0$ top-stabilizable terms. The basic observation is that if a top-stabilizable constant, say $c$, occurs at a certain position in a term, say $fct_2 \ldots t_m$, then this term $(fct_2 \ldots t_m)$ is equivalent to a term $t = ft_1t_2 \ldots t_m$ with the property that $t$ rewrites to a depth $0$ term via $R$ only if $fxt_2 \ldots t_m$ also does. Here, $t_1$ is chosen to be top-stable. So, when considering rewrites on $fct_2 \ldots t_m$ or $ft_1 \ldots t_m$, we should treat $c$ and $t_1$ as variables. This is roughly the intuition behind the following definitions of marked terms and marked rewriting.

**Definition 6.** *A marking $M$ of a term $t$ is a set of leaf positions in $t$. A term $t$ with a marking $M$ is denoted by $(t, M)$.*

*A marked term $(s, M)$ rewrites to $(t, N)$ via marked rewriting if $s \to_{l \to r \in R, p} t$ for some position $p \notin M$ such that, if $l|_{p_1}$ is a constant then $p.p_1 \notin M$, and the new marking $N$ satisfies: (a) for all $q$ disjoint with $p$, we have $q \in M$ iff $q \in N$, (b) for all $p_1$, $p_2$ and $q$ such that $l|_{p_1}$ and $r|_{p_2}$ are the same variable, $p.p_1.q \in M$ iff $p.p_2.q \in N$, and (c) no more positions are in $N$.*

*A marked flat term $(s = f\alpha_1 \ldots \alpha_m, M)$ is said to be correctly marked if for all $i \in M$, we have that $\alpha_i$ is top-stabilizable.*

*Example 2.* The marked term $(0 + x, \{1\})$, denoted as $\underline{0} + x$, cannot be rewritten with the rule $0 + x \to x$, but it can be rewritten with the rule $x + y \to y + x$ to $x + \underline{0}$.

The notions of size-irreducible terms, increasing derivations and top-stable terms can be adapted naturally to marked terms. All the arguments of Lemmas 1 and 2 are also valid for marked rewriting, and we have:

**Lemma 4.** *If $(s, M) \to_R^* (t, N)$, then there is a derivation of the form $(s, M) \to_F^* \circ \to_P^* \circ \to_B^* (t, N)$. If $(s, M)$ is size-irreducible and $(s, M) \to_R^* (t, N)$, then there is an increasing derivation $(s, M) \to_R^* (t, N)$.*

### 2.5 The Sets $S_\infty$ and $J_\infty$

The set $S_0$ of top-stable flat terms can be extended with empty markings to give the new set

$$\{(f\alpha_1 \ldots \alpha_m, \emptyset) : f\alpha_1 \ldots \alpha_m \in S_0\}$$

of marked top-stable terms, which we also denote by $S_0$. We add new marked flat terms to this set to get the set of all correctly marked top-stable flat terms and top-stabilizable constants (and a variable if some variable is top-stabilizable) using the following fixpoint computation, starting with the new set $S_0$.

$$\begin{aligned} S_{j+1} = S_j &\cup \{c : c \leftrightarrow^* f\alpha_1 \ldots \alpha_m \text{ for some } (f\alpha_1 \ldots \alpha_m, M) \in S_j\} \\ &\cup \{(f\alpha_1 \ldots \alpha_m, M) : (f\alpha_1 \ldots \alpha_m, M) \text{ is top-stable and } \forall i \in M : \alpha_i \in S_j\} \end{aligned}$$

Note that by Lemma 4, $(f\alpha_1 \ldots \alpha_m, M)$ is top-stable iff it is irreducible by $F$ by marked rewriting.

This iterative procedure of computing larger and larger subsets $S_j$ of the set of all marked flat terms is guaranteed to terminate in a polynomial number of steps. This is because the total number of flat marked terms, up to variable renaming, is polynomial, assuming $m$ is a constant.

**Lemma 5.** *If $S_\infty$ is the fixpoint of the computation above, then, up to variable renaming, $(f\alpha_1 \ldots \alpha_m, M) \in S_\infty$ iff $(f\alpha_1 \ldots \alpha_m, M)$ is top-stable and correctly marked, and a depth $0$ term $c \in S_\infty$ iff $c$ is top-stabilizable.*

**Definition 7.** *Two marked terms $(s, M)$ and $(t, N)$ are said to be structurally joinable if $(s, M) \to_R^* (s', M')$ and $(t, N) \to_R^* (t', N')$ for some terms $s'$ and $t'$ with the same structure (i.e., $Pos(s') = Pos(t')$, where $Pos(s')$ is the set of all positions in $s'$) and equivalent leaf terms (i.e., for all leaf positions[1] $p \in Pos(s')$, we have that $s'|_p$ and $t'|_p$ are equivalent).*

We use the following fixpoint computation to obtain some structurally joinable pairs of marked terms.

$$\begin{aligned} J_0 &= \{((\alpha, \emptyset), (\beta, M)) : \alpha \leftrightarrow_R^* \beta \text{ and } \alpha, \beta \text{ are depth 0 terms}\} \\ J_{j+1} &= J_j \cup \{((\alpha, \emptyset), (f\beta_1 \ldots \beta_m, M)) : \\ &\qquad (f\beta_1 \ldots \beta_m, M) \text{ is top-stable,} \\ &\qquad \alpha \to_R^r fa_1 \ldots a_m, (f\beta_1 \ldots \beta_m, M) \to_R^r (fb_1 \ldots b_m, N), \text{ and} \\ &\qquad \forall i \in \{1 \ldots m\} \text{ either } a_i = b_i \text{ or } ((a_i, \emptyset), (b_i, N|_i)) \in J_j\} \end{aligned}$$

where $N|_i$ contains the positions $p$ such that $i.p \in N$. Note that the $b_i$ can be considered depth 0 or 1 terms, and that the $a_i$ can be considered depth 0 or satisfying $a_i = b_i$.

**Lemma 6.** *If $J_\infty$ is the fixpoint of above computation, then it is the set of all structurally joinable pairs of terms of the form $((\alpha, \emptyset), (\beta, M))$ or $((\alpha, \emptyset), (f\beta_1 \ldots \beta_m, M))$, where $\alpha, \beta$ are depth $0$ terms, and $(f\beta_1 \ldots \beta_m, M)$ is a flat top-stable marked term.*

[1] Note that due to Proposition 1, for non-leaf positions $p \in Pos(s')$, $s'|_p = t'|_p = f$.

### 2.6 The Technical Lemma and the Result

**Definition 8.** *A pair of rules* $((l \to r), (l' \to r'))$ *is* useless *if* $l = x$ *and* $l' = y$ *for some variables* $x$ *and* $y$ *that appear in* $r$ *and* $r'$*, respectively, at the same non-root position.*

*Two top-stable marked flat terms* $(f\alpha_1 \dots \alpha_m, M)$ *and* $(f\alpha'_1 \dots \alpha'_m, M')$ *are* first-step joinable *if there exist*

$$(f\alpha_1 \dots \alpha_m, M) \to^r_{(l \to r)\sigma} (fs_1 \dots s_m, N)$$
$$(f\alpha'_1 \dots \alpha'_m, M') \to^r_{(l' \to r')\theta} (fs'_1 \dots s'_m, N')$$

*such that every* $s_i$ *is equivalent to its corresponding* $s'_i$*, and* $((l \to r), (l' \to r'))$ *is not useless.*

Note that first-step joinability can be efficiently computed, since it is enough to consider subterms $s_i$ and $s'_i$ of depth 0 or 1: if $r|_i$ is a variable not in $l$, we can force $r|_i\sigma = r'|_i\theta$ by modifying the substitutions, and the same if $r'|_i$ is a variable not in $l'$.

The polynomial time test for confluence depends on the following characterization using the sets $S_\infty$ and $J_\infty$ and the notion of first-step joinability.

**Lemma 7.** *The rewrite system* $R$ *is confluent if, and only if,*

*(c1) Every pair* $\alpha, \beta$ *of equivalent depth* $0$ *terms is joinable,*
*(c2) If* $\alpha \leftrightarrow^* f\beta_1 \dots \beta_m$ *and* $(f\beta_1 \dots \beta_m, M) \in S_\infty$ *then* $(\alpha, \emptyset)$ *and* $(f\beta_1 \dots \beta_m, M)$ *are structurally joinable, i.e.* $((\alpha, \emptyset), (f\beta_1 \dots \beta_m, M)) \in J_\infty$
*(c3) If* $(f\alpha_1 \dots \alpha_m, M) \in S_\infty$ *and* $(f\beta_1 \dots \beta_m, N) \in S_\infty$ *are such that* $f\alpha_1 \dots \alpha_m \leftrightarrow^* f\beta_1 \dots \beta_m$*, then these two marked terms in* $S_\infty$ *are first-step joinable.*

*Proof.* (Sketch) A correctly marked flat term $(f\alpha_1 \dots \alpha_m, M)$ can be *lifted* to a term $fs_1 \dots s_m$ by replacing the marked $\alpha_i$'s by equivalent top-stable and $F$-irreducible terms $s_i$'s.

$\Rightarrow$: Suppose $R$ is confluent. Condition (c1) follows from the definition of confluence.

*Condition (c2).* Suppose $\alpha \leftrightarrow^* f\beta_1 \dots \beta_m$ and $(f\beta_1 \dots \beta_m, M) \in S_\infty$. Using Lemma 5, we can lift $(f\beta_1 \dots \beta_m, M)$ to the term $ft_1 \dots t_m$. Since $(f\beta_1 \dots \beta_m, M)$ is top-stable, it follows that $ft_1 \dots t_m$ is top-stable. Now, $\alpha$ is equivalent to $ft_1 \dots t_m$ and by confluence they are joinable. Since both are size-irreducible, there are increasing derivations of the form $\alpha \to^*_R u$ and $ft_1 \dots t_m \to^*_R u$. We can extract an increasing derivation $(f\beta_1 \dots \beta_m, M) \to^*_R (u' = u[\beta'_1]_{p_1} \dots [\beta'_k]_{p_k}, N = \{p_1, \dots, p_k\})$ from the latter derivation by ignoring all rewrite steps at or below marked positions. Using an auxiliary lemma, we can show that there exists a derivation $\alpha \to^*_R u[\beta''_1]_{p_1} \dots [\beta''_m]_{p_k}$, such that $\beta''_i \leftrightarrow^*_R \beta'_i$.

*Condition (c3).* Let $(f\alpha_1 \dots \alpha_m, M)$ and $(f\beta_1 \dots \beta_m, N)$ be marked flat terms in $S_\infty$ such that $f\alpha_1 \dots \alpha_m \leftrightarrow^*_R f\beta_1 \dots \beta_m$. Again, using Lemma 5, we can lift $(f\alpha_1 \dots \alpha_m, M)$ and $(f\beta_1 \dots \beta_m, N)$ to size-irreducible terms $s = fs_1 \dots s_m$ and $t = ft_1 \dots t_m$. By confluence, $s$ and $t$ are joinable, and hence there exist increasing derivations $fs_1 \dots s_m \to^r_{(l \to r)\sigma} fs'_1 \dots s'_m \to^{*,nr}_R u$ and $ft_1 \dots t_m \to^r_{(l' \to r')\theta} ft'_1 \dots t'_m \to^{*,nr}_R u$. Such a $u$ can be chosen minimally, and consequently the pair $(l \to r, l' \to r')$ is not useless. Clearly, every $s'_i$ is equivalent to the corresponding $t'_i$.

Now, by suitably modifying the substitutions $\sigma$ and $\theta$, we can get marked rewrite steps, $(f\alpha_1 \ldots \alpha_m, M) \to_{(l\to r)\sigma'} (fs''_1 \ldots s''_m, M')$ and $(f\beta_1 \ldots \beta_m, N) \to_{(l'\to r')\theta'} (ft''_1 \ldots t''_m, N')$, such that $s''_i \leftrightarrow^*_R s'_i \leftrightarrow^*_R t'_i \leftrightarrow^*_R t''_i$. This shows that the two marked terms $(f\alpha_1 \ldots \alpha_m, M)$ and $(f\beta_1 \ldots \beta_m, N)$ are first-step joinable.

$\Leftarrow$: Suppose conditions (c1), (c2), and (c3) are satisfied, but $R$ is not confluent. Let $\{s, t\}$ be a witness to non-confluence, that is, $s$ and $t$ are equivalent, but not joinable. We compare witnesses by a multiset extension of the ordering $\succ$ defined earlier. First, we note that both $s$ and $t$ can be assumed to be size-irreducible, otherwise we would have a smaller counterexample to confluence.

If $s = fs_1 \ldots s_m$, then each $s_i$ is either top-stable or of depth $0$. Similarly, for the term $t$. Additionally, if the top-stable subterms $s_i$ are equivalent to some depth 0 terms, then the term $s$ can be *projected* onto a correctly marked flat term $(f\alpha_1 \ldots \alpha_m, M)$ where either $\alpha_i$ is the depth 0 term equivalent to $s_i$ and $i \in M$, or $\alpha_i = s_i$. We differentiate the following cases based on the form of $s$ and $t$:

*Case 1. $s$ and $t$ are both depth $0$ terms*: In this case, Condition (c1) implies that $s$ and $t$ are joinable, a contradiction.

*Case 2. $s$ is a depth $0$ term $\alpha$ and $t = ft_1 \ldots t_m$*: We first claim that each $t_i$ is equivalent to a depth 0 term. If not, then w.l.o.g. let $t_1$ not be equivalent to any depth 0 term. Then $t_1$ cannot be "used" in the proof $\alpha \leftrightarrow^*_R ft_1 \ldots t_m$, and hence it can be replaced by a new variable $x$ in this proof to yield a new proof $\alpha' \leftrightarrow^*_R fxt_2 \ldots t_m$. If $\alpha'$ and $fxt_1 \ldots t_m$ are not joinable, then they are a smaller witness to non-confluence, a contradiction. If $\alpha'$ and $fxt_1 \ldots t_m$ are joinable, then $\alpha' = x$, and $\alpha$ and $t_1$ are equivalent, but not joinable. The pair $\{\alpha, t_1\}$ is a smaller witness to non-confluence, a contradiction again.

Let $(f\beta_1 \ldots \beta_m, M)$ be a projection of $t$. This marked term is top-stable and correctly marked, and hence by Lemma 5, it is in $S_\infty$. By condition (c2), $(\alpha, \emptyset)$ and $(f\beta_1 \ldots \beta_m, M)$ are structurally joinable, and hence, there exist $(\alpha, \emptyset) \to^*_R (s', \emptyset)$ and $(f\beta_1 \ldots \beta_m, M) \to^*_R (t', M')$ such that $Pos(s') = Pos(t')$, and for every leaf position $p \in Pos(s')$ we have $s'|_p \leftrightarrow^*_R t'|_p$. If $M' = \{p_1 \ldots p_k\}$, then $t' = t'[\beta|_{i_1}]_{p_1} \ldots [\beta|_{i_k}]_{p_k}$, for some $i_1 \ldots i_k \subseteq M$.

If we mimic the derivation $f\beta_1 \ldots \beta_m \to^*_R t'$, but now starting from $ft_1 \ldots t_m$, we obtain a derivation of the form $ft_1 \ldots t_m \to^*_R t'' = t'[t_{i_1}]_{p_1} \ldots [t_{i_k}]_{p_k}$. Moreover, each $t_{i_j}$ is equivalent to $t'|_{p_j} = \beta_{i_j}$, and hence, for each leaf position $p$ of $s'$ we have that $s'|_p$ and $t''|_p$ are equivalent, and $size(t''|_p) < size(t)$. Since $\alpha$ and $ft_1 \ldots t_m$ are not joinable, $s'$ and $t''$ are not joinable, and hence, for some leaf position $p$ of $s'$ we have that $s'|_p$ and $t''|_p$ are not joinable. For such a $p$, $\{s'|_p, t''|_p\}$ is a smaller witness to non-confluence, a contradiction.

*Case 3. $s = fs_1 \ldots s_m$ and $t = ft_1 \ldots t_m$*: Using arguments similar to the previous case, we can assume that the $s_i$'s and $t_i$'s are equivalent to depth 0 terms. Let $(f\alpha_1 \ldots \alpha_m, M)$ and $(f\beta_1 \ldots \beta_m, N)$ be the projections of $s$ and $t$. Both these marked terms are top-stable and correctly marked and hence, by Lemma 5, they are in $S_\infty$. By condition (c3), they are first-step joinable, i.e. there exist $(f\alpha_1 \ldots \alpha_m, M) \to^r_{(l_1\to r_1)\sigma} (fs'_1 \ldots s'_m, M')$ and $(f\beta_1 \ldots \beta_m, N) \to^r_{(l_2\to r_2)\theta} ft'_1 \ldots t'_m, N')$ such that every $s'_i$ is equivalent to its corresponding $t'_i$, and $((l_1 \to r_1), (l_2 \to r_2))$ is not useless.

We apply these rewrite steps to the original terms to get $fs_1 \dots s_m \to^r_{(l_1 \to r_1)\sigma'} fs''_1 \dots s''_m = s''$ and $ft_1 \dots t_m \to^r_{(l_2 \to r_2)\theta'} ft''_1 \dots t''_m = t''$, by choosing $\sigma'$ and $\theta'$ such that for each $i \in \{1 \dots m\}$, it is the case that (a) if $r_1|_i$ ($r_2|_i$) is a variable not appearing in $l_1$ ($l_2$), then $s''_i = t''_i$, and (b) if not, then either $s''_i$ ($t''_i$) is a constant or it coincides with one of the $s_j$ ($t_j$) or it coincides with $s$ ($t$). But it cannot happen that both $s''_i$ and $t''_i$ coincide with $s$ and $t$, respectively. This is because the rules $l_1 \to r_1$ and $l_2 \to r_2$ are not useless.

By construction, every $s''_i$ is equivalent to its corresponding $t''_i$. Since $s$ and $t$ are not joinable, $s''$ and $t''$ are not joinable, and hence, for some $i \in \{1 \dots m\}$ we have that $s''_i$ and $t''_i$ are not joinable. This can only happen for case (b) above, and by the previous observation, $(s''_i, t''_i)$ is a smaller witness to non-confluence, a contradiction.

Finally, we are ready to state the main result.

**Theorem 1.** *Confluence of linear shallow term rewrite systems can be decided in time polynomial in the size of the rewrite system, assuming the maximum arity of any function symbol is bounded by a constant.*

*Proof.* The input linear shallow rewrite system is transformed into a flat linear rewrite system and then it is saturated under the ordered chaining inference rules. Flattening increases the size $|\mathcal{F}|$ of the signature by a linear factor of the input size. Now, the number of flat linear rewrite rules is bounded by a polynomial in the size $|\mathcal{F}|$ of the signature, and hence these two transformation steps run in polynomial time. Next, the sets $S_\infty$ and $J_\infty$ are computed, again using polynomial time fixpoint computations. Finally, confluence is tested using the characterization given in Lemma 7. The three conditions in Lemma 7 can be tested in polynomial time: (a) Equivalent depth zero terms can be identified because equivalence testing for flat linear rewrite systems can be efficiently done, say using standard completion modulo permutation rules. Joinability of depth zero terms can be tested in polynomial time using simple fixpoint computations, similar to previous work [13]. (b) It is also clear that the conditions (c2) and (c3) can be tested in polynomial time.

*Example 3.* For the rewrite system $R$ of Example 1, the set $S_\infty$ contains the terms $x + y, 0 + x, x + 0, 0 + 0$, where the positions of 0 are marked. But the pairs $(0, 0 + 0), (x, 0 + x), (x, x + 0)$ are easily seen to be structurally joinable, while $(x + 0, 0 + x)$ is first-step joinable. Hence, this rewrite system is confluent.

## 3 Relaxing the Restrictions

The reachability, 2-joinability, and confluence problems for shallow term rewrite systems are not known to be decidable. But, we can establish the following lower-bounds.

**Theorem 2.** *The reachability problem for shallow term rewrite systems is* EXPTIME-*hard, even when the maximum arity is a constant.*

*Proof.* We reduce the problem of deciding non-emptiness of language intersection of $n$ bottom-up tree-automata to this problem. The proof is similar to the proof of EXPTIME-hardness of rigid-reachability of shallow terms over ground systems [5]. Let $R_1$ be the union of the *reversed* transitions of all the $n$ tree-automata. We assume that the tree-automata have disjoint states with accepting states $q_1, q_2, \ldots, q_n$, respectively. Let $R_2 = \{a \to g(q_1, f(q_2, f(q_3, \cdots, f(q_{n-1}, q_n) \cdots))), fxx \to x, gxx \to b\}$, where $g, f$ are two new binary function symbols and $a, b$ are two new constants. Now, $a$ rewrites to $b$ via $R_1 \cup R_2$ iff the intersection of languages accepted by the $n$ automata is nonempty.

For shallow term rewrite systems, EXPTIME-hardness of 2-joinability follows from the hardness of reachability using the reduction in [15]. We next show hardness of deciding confluence of shallow term rewrite systems by modifying the proof of Theorem 2.

**Theorem 3.** *Deciding confluence of shallow term rewrite systems is EXPTIME-hard, even when the maximum arity is a constant.*

*Proof.* We add additional rewrite rules to the rewrite system generated in the proof of Theorem 2 to make the system confluent exactly when $b$ is reachable from $a$. First, we introduce a new constant $c$ in the signature $\mathcal{G}$ of the tree automata and convert all constants $d$ to unary terms $d(c)$. The rules in $R_1$ are modified to reflect this change. We assume that some ground term can be reached from any tree-automata state $q$ via $R_1$. Let $R_3 = \{c \to a,\ h(x_1, \ldots, x_{i-1}, b, x_{i+1}, \ldots, x_n) \to b \textit{ for all } h \in \mathcal{G},\ fxb \to b,\ fbx \to b,\ gxb \to b,\ gbx \to b\}$. Now, consider the shallow term rewrite system $R = R_1 \cup R_2 \cup R_3 \cup \{d \to a, d \to b\}$, where $R_1$ and $R_2$ are as in proof of Theorem 2 and $d$ is a new constant in the signature. We claim without proof that $R$ is confluent iff the $n$ tree-automata have a non-empty language intersection.

We also note here that reachability, 2-joinability, and confluence problems are undecidable for linear (non-shallow) term rewrite systems [15].

## 4 Conclusion

In this paper we presented a polynomial time algorithm for deciding confluence of linear shallow term rewrite systems where each variable is allowed at most two occurrences in a rule—one on each side. The time complexity analysis assumes that the maximum arity of a function symbol in the signature is a constant. Our result generalizes those in [2, 13]. We also show that the reachability, joinability and confluence problems are all EXPTIME-hard for shallow non-linear systems, and all three are known to be undecidable for linear non-shallow systems, which indicates that our assumptions can not be easily relaxed without considerably losing efficiency. Our technique can be adapted to decide ground confluence of linear shallow term rewrite systems in polynomial time. It is not clear whether our method can give polynomial time algorithms to decide confluence when we have non-fixed arity or for rule-linear rewrite systems (no variable appears twice in the whole rule) and not necessarily shallow, and this is a matter for future work.

**Acknowledgments.** We would like to thank the reviewers for their helpful comments.

## References


1. L. Bachmair. *Canonical Equational Proofs*. Birkhäuser, Boston, 1991.
2. H. Comon, G. Godoy, and R. Nieuwenhuis. The confluence of ground term rewrite systems is decidable in polynomial time. In *42nd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, Las Vegas, Nevada, USA, 2001.
3. H. Comon, M. Haberstrau, and J.-P. Jouannaud. Syntacticness, cycle-syntacticness, and shallow theories. *Information and Computation*, 111(1):154–191, 1994.
4. M. Dauchet, T. Heuillard, P. Lescanne, and S. Tison. Decidability of the confluence of finite ground term rewrite systems and of other related term rewrite systems. *Information and Computation*, 88(2):187–201, October 1990.
5. H. Ganzinger, F. Jacquemard, and M. Veanes. Rigid reachability: The non-symmetric form of rigid E-unification. *Intl. Journal of Foundations of Computer Science*, 11(1):3–27, 2000.
6. Guillem Godoy, Robert Nieuwenhuis, and Ashish Tiwari. Classes of Term Rewrite Systems with Polynomial Confluence Problems. *ACM Transactions on Computational Logic (TOCL)*, 2002. To appear.
7. A. Hayrapetyan and R.M. Verma. On the complexity of confluence for ground rewrite systems. In *Bar-Ilan International Symposium On The Foundations Of Artificial Intelligence*, 2001. Proceedings on the web at `http://www.math.tau.ac.il/~nachumd/bisfai-pgm.html`.
8. D. E. Knuth and P. B. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational Problems in Abstract Algebra*, pages 263–297. Pergamon Press, Oxford, 1970.
9. A. Levy and J. Agusti. Bi-rewriting, a term rewriting technique for monotone order relations. In C. Kirchner, editor, *Rewriting Techniques and Applications RTA-93*, pages 17–31, 1993. LNCS 690.
10. R. Nieuwenhuis. Basic paramodulation and decidable theories. In *11th IEEE Symposium on Logic in Computer Science, LICS 1996*, pages 473–482. IEEE Computer Society, 1996.
11. M. Oyamaguchi. The Church-Rosser property for ground term-rewriting systems is decidable. *Theoretical Computer Science*, 49(1):43–79, 1987.
12. A. Tiwari. Rewrite closure for ground and cancellative AC theories. In R. Hariharan and V. Vinay, editors, *Conference on Foundations of Software Technology and Theoretical Computer Science, FST&TCS '2001*, pages 334–346. Springer-Verlag, 2001. LNCS 2245.
13. A. Tiwari. Deciding confluence of certain term rewriting systems in polynomial time. In Gordon Plotkin, editor, *IEEE Symposium on Logic in Computer Science, LICS 2002*, pages 447–456. IEEE Society, 2002.
14. A. Tiwari. *On the combination of equational and rewrite theories induced by certain term rewrite systems*. Menlo Park, CA 94025, 2002. Available at: `www.csl.sri.com/~tiwari/combinationER.ps`.
15. R. Verma, M. Rusinowitch, and D. Lugiez. Algorithms and reductions for rewriting problems. *Fundamenta Informaticae*, 43(3):257–276, 2001. Also in Proc. of Int'l Conf. on Rewriting Techniques and Applications 1998.
16. H. Zantema. Termination of term rewriting: interpretation and type elimination. *Journal of Symbolic Computation*, 17:23–50, 1994.