Journal of Universal Computer Science, vol. 3, no. 11 (1997), 1134-1147
submitted: 8/8/97, accepted: 15/10/97, appeared: 28/11/97 © Springer Pub. Co.

Persistency of Confluence[1]

Takahito Aoto
(JAIST, Japan
aoto@jaist.ac.jp)

Yoshihito Toyama
(JAIST, Japan
toyama@jaist.ac.jp)

Abstract: A property $\mathcal{P}$ of term rewriting systems (TRSs, for short) is said to be persistent if for any many-sorted TRS $\mathcal{R}$, $\mathcal{R}$ has the property $\mathcal{P}$ if and only if its underlying unsorted TRS $\Theta(\mathcal{R})$ has the property $\mathcal{P}$. This notion was introduced by H. Zantema (1994). In this paper, it is shown that confluence is persistent.

Key Words: term rewriting system, persistency, confluence, modularity
Category: F4.1

1 Introduction

Confluence is a property that is widely studied in the theory of term rewriting [Knuth and Bendix 1970][Huet 1980][Toyama 1988]. One of the approaches to detect the confluence of a given term rewriting system (TRS, for short) is to infer it from those of its subsystems. It was shown in [Toyama 1987] that the confluence of *the direct sum of TRSs* can be inferred from those of its components. Here, the (disjoint) union of TRSs $\mathcal{R}_1$ and $\mathcal{R}_2$ is said to be the direct sum of $\mathcal{R}_1$ and $\mathcal{R}_2$ when the sets of function symbols that appear in $\mathcal{R}_1$ and $\mathcal{R}_2$ are disjoint. Following standard terminology, we say "a property $\mathcal{P}$ is modular" if $\mathcal{P}$ is inferred from those of its subsystems. For confluence, unlike for other important properties of TRSs, only few modularity results which relax this "direct sum" limitation are known [Ohlebusch 1995][Kitahara 1995].

A property $\mathcal{P}$ of TRSs is said to be *persistent* if for any many-sorted TRS $\mathcal{R}$, $\mathcal{R}$ has the property $\mathcal{P}$ if and only if its underlying unsorted TRS $\Theta(\mathcal{R})$ has the property $\mathcal{P}$. This notion is due to H. Zantema [Zantema 1994]. He showed that every component-closed property $\mathcal{P}$ of TRSs is modular for the direct sum of TRSs whenever $\mathcal{P}$ is persistent; however, we note that many properties of TRSs including termination, confluence, etc. are component-closed, and hence, in particular, the persistency of confluence generalizes the modularity of confluence with respect to the direct sum. Also he showed that termination is persistent for non-collapsing TRSs and non-duplicating TRSs, generalizing a similar result [Rusinowitch 1987] for the modularity of termination with respect to the direct sum. Further investigations on the persistent properties related to termination such as that of termination modulo equations, etc. also appeared in [Ohsaki and Middeldorp 1997].

[1] Proceedings of the *First Japan-New Zealand Workshop on Logic in Computer Science*, special issue editors D.S. Bridges, C.S. Calude, M.J. Dinneen and B. Khoussainov.

In this paper, we show the persistency of confluence, generalizing a similar result [Toyama 1987] for the modularity of confluence with respect to the direct sum. Although a proof of the persistency of confluence was first appeared in the previous version [Aoto and Toyama 1996] of this paper, the persistency of confluence had been conjectured by H. Zantema, Y. Toyama and A. Middeldorp in a private communication. It is also mentioned in [Zantema 1994] that a straightforward modification of the proof in [Toyama 1987] seems applicable to show the persistency of confluence. In this paper, however, we rather show it via a straightforward modification of a simplified proof appeared in [Klop et al. 1994].

Persistency of confluence can be used to infer the confluence of a given TRS from those of its subsystems; this approach also relaxes the direct sum condition [see Example 1]. As proved in [Aoto and Toyama 1997], however, this kind of application of persistency is successful also for other properties that are modular for the direct sum, e.g. UN [Middeldorp 1989a], NF for left-linear TRSs [Middeldorp 1990], $\mathrm{UN}^{\rightarrow}$ for for left-linear TRSs [Marchiori 1996] and those appeared in [Gramlich 1995], [Middeldorp 1989b], [Schmidt et al. 1995], [Toyama et al. 1995]. Nevertheless, it is unknown whether these properties are persistent. In [van de Pol 1993] it is shown that for any component-closed property $\mathcal{P}$ of TRSs, $\mathcal{P}$ is persistent if and only if $\mathcal{P}$ is modular for the direct sum of many-sorted TRSs. Thus, our result immediately gives the modularity of confluence with respect to the direct sum of many-sorted TRSs.

2 Preliminaries

2.1 Sorted term rewriting systems

In this subsection, we review some basic notions of sorted term rewriting and fix the notations that will be used in this paper.

Our language is given by a set $\mathcal{S}$ of sorts (denoted by $\alpha, \beta, \gamma, \ldots$), a set $\mathcal{V}$ of variables (denoted by $x, y, z, \ldots$), and a set $\mathcal{F}$ of function symbols (denoted by $f, g, h, \ldots$). Each variable is given with its sort; we assume that there are countably infinite variables of sort α for each sort $\alpha \in \mathcal{S}$. Similarly, each function symbol is given with the sorts of its arguments and the sort of its output. We write $f : \alpha_1 \times \cdots \times \alpha_n \rightarrow \beta$ if f takes n arguments of sorts $\alpha_1, \ldots, \alpha_n$ respectively to a value of sort β. A function symbol with no arguments is called a *constant*.

With such language, one can build up terms (of sort α) in a usual way: (1) a variable of sort α is a term of sort α; (2) if $f : \alpha_1 \times \cdots \times \alpha_n \rightarrow \alpha$ is a function symbol and $t_1, \ldots, t_n$ are terms of sort $\alpha_1, \ldots, \alpha_n$ respectively, then $f(t_1, \ldots, t_n)$ is a term of sort α. Let $\mathcal{T}$ (and $\mathcal{T}^\alpha$) denote the set of terms (of sort α, respectively). We also write "$t : \alpha$" to indicate that $t \in \mathcal{T}^\alpha$. Syntactical equality is denoted by $\equiv$. $\mathcal{V}(t)$ is the set of variables that appear in a term t.

For each sort α, let $\Box^\alpha$ be a special constant—called a *hole*—of sort α. A *context* is a term possibly containing holes. The set of contexts is denoted by $\mathcal{C}$. We write $C : \alpha_1 \times \cdots \times \alpha_n \rightarrow \alpha$ when $C \in \mathcal{C}$ has the sort α (as a term) and has n holes $\Box^{\alpha_1}, \ldots, \Box^{\alpha_n}$ from left to right in it. If $C : \alpha_1 \times \cdots \times \alpha_n \rightarrow \beta$ and $t_1 : \alpha_1, \ldots, t_n : \alpha_n$ then $C[t_1, \ldots, t_n]$ is the term obtained from C by replacing holes with $t_1, \ldots, t_n$ from left to right. A context C is written as $C[\,]$ when C contains precisely one hole. A term t is said to be a *subterm* of s ($t \trianglelefteq s$, in symbol) if $s \equiv C[t]$ for some context $C[\,]$.

A *substitution* σ is a mapping from $\mathcal{V}$ to $\mathcal{T}$ such that x and $\sigma(x)$ have the same sort. A substitution is extended to a homomorphism from $\mathcal{T}$ to $\mathcal{T}$ in an obvious way. For a substitution σ and a term t, we customarily write $t\sigma$ instead of $\sigma(t)$.

A *(many-sorted) rewrite rule* is a pair $\langle l, r\rangle$ of terms such that (1) l and r have the same sort, (2) $l \notin \mathcal{V}$, (3) $\mathcal{V}(r) \subseteq \mathcal{V}(l)$. We conventionally write $l \to r$ instead of $\langle l, r\rangle$. A rewrite rule $l \to r$ is *collapsing* if $r \in \mathcal{V}$. A *many-sorted term rewriting system* (*STRS*, for short) is a set of rewrite rules.

Given a STRS $\mathcal{R}$, a term s *reduces* to a term t ($s \to_{\mathcal{R}} t$, in symbol) when $s \equiv C[l\sigma]$ and $t \equiv C[r\sigma]$ for some $C[\ ] \in \mathcal{C}$, $l \to r \in \mathcal{R}$ and substitution σ. We call $s \to_{\mathcal{R}} t$ a *rewrite step* (or a *reduction*). The *redex* of this rewrite step is $l\sigma$. The term t is called a *reduct* of the term s. One can readily check that s and t have the same sort whenever $s \to_{\mathcal{R}} t$.

The converse relation, the symmetric closure and the reflexive closure of $\to_{\mathcal{R}}$ are denoted by $\leftarrow_{\mathcal{R}}$, $\leftrightarrow_{\mathcal{R}}$ and $\xrightarrow{\equiv}_{\mathcal{R}}$, respectively. The transitive reflexive closures of $\to_{\mathcal{R}}$, $\leftarrow_{\mathcal{R}}$ and $\leftrightarrow_{\mathcal{R}}$ are denoted by $\xrightarrow{*}_{\mathcal{R}}$, $\xleftarrow{*}_{\mathcal{R}}$ and $\xleftrightarrow{*}_{\mathcal{R}}$, respectively. Terms t_1 and t_2 are *joinable* if there exists some term t' such that $t_1 \xrightarrow{*}_{\mathcal{R}} t' \xleftarrow{*}_{\mathcal{R}} t_2$. A term t is *confluent* if for any terms t_1 and t_2, t_1 and t_2 are joinable whenever $t_1 \xleftarrow{*}_{\mathcal{R}} t \xrightarrow{*}_{\mathcal{R}} t_2$. A STRS $\mathcal{R}$ is confluent if every term is confluent with respect to its reduction $\to_{\mathcal{R}}$. Henceforth, the subscript $_{\mathcal{R}}$ will be omitted when $\mathcal{R}$ is obvious from the context.

When $\mathcal{S} = \{*\}$, a STRS is called a term rewriting system (TRS, for short). Given an arbitrary STRS $\mathcal{R}$, by identifying each sort with $*$, we obviously obtain a TRS $\Theta(\mathcal{R})$—called the underlying TRS of $\mathcal{R}$. The following definition is due to [Zantema 1994].

Definition 1. A property $\mathcal{P}$ of STRSs is said to be *persistent* if

$$\mathcal{R} \text{ has the property } \mathcal{P} \Leftrightarrow \Theta(\mathcal{R}) \text{ has the property } \mathcal{P}.$$

2.2 Sorting of term rewriting systems

In this subsection, we introduce the notion of *sort attachment* to TRSs.

Let $\mathcal{F}$ and $\mathcal{V}$ be sets of function symbols and variables, respectively, on a trivial set $\{*\}$ of sorts. Terms built from this language are called *unsorted terms.* Let $\mathcal{S}$ be another set of sorts. A *sort attachment* τ *on* $\mathcal{S}$ is a mapping from $\mathcal{F} \cup \mathcal{V}$ to the set $\mathcal{S}^*$ of finite sequences of elements from $\mathcal{S}$ such that $\tau(x) \in \mathcal{S}$ for any $x \in \mathcal{V}$ and $\tau(f) \in \mathcal{S}^{n+1}$ for any n-ary function symbol $f \in \mathcal{F}$. We write $\tau(f) = \alpha_1 \times \cdots \times \alpha_n \to \beta$ instead of $\tau(f) = \alpha_1, \ldots, \alpha_n, \beta$. Without loss of generality we assume that there are countably infinite variables x with $\tau(x) = \alpha$ for each $\alpha \in \mathcal{S}$. The set of τ-sorted function symbols from $\mathcal{F}$ is denoted by $\mathcal{F}^\tau$.

A term t is said to be *well-sorted under* τ *with sort* α if $\vdash_\tau t : \alpha$ is derivable in the following inference system:

$$\frac{\tau(x) = \alpha}{\vdash_\tau x : \alpha} \tag{1}$$

$$\frac{\tau(f) = \alpha_1 \times \cdots \times \alpha_n \to \beta \quad \vdash_\tau t_1 : \alpha_1 \quad \cdots \quad \vdash_\tau t_n : \alpha_n}{\vdash_\tau f(t_1, \ldots, t_n) : \beta.} \tag{2}$$

The set of well-sorted terms under τ is denoted by $\mathcal{T}^\tau$, i.e. $\mathcal{T}^\tau = \{t \in \mathcal{T} \mid \vdash_\tau t : \alpha \text{ for some } \alpha \in \mathcal{S}\}$. Clearly, $\mathcal{T}^\tau \subseteq \mathcal{T}$. For a context $C \in \mathcal{C}$, we write $C : \alpha_1 \times \cdots \times \alpha_n \to \beta$ if $\vdash_\tau C[\Box^{\alpha_1}, \ldots, \Box^{\alpha_n}] : \beta$ is derivable by rules (1), (2) with an additional rule:

$$\frac{\alpha \in \mathcal{S}}{\vdash_\tau \Box^\alpha : \alpha.} \tag{3}$$

Let $\mathcal{R}$ be a TRS. It may be assumed that the variables have been renamed appropriately so that every two rewrite rules in $\mathcal{R}$ share no variables. A sort attachment τ is said to be *consistent with* $\mathcal{R}$ if for any $l \to r \in \mathcal{R}$, l and r are well-sorted under τ with the same sort.

From a given TRS $\mathcal{R}$ and a sort attachment τ consistent with $\mathcal{R}$, by regarding each function symbol f to be of sort $\tau(f)$, and each variable x to be of sort $\tau(x)$, we get a STRS $\mathcal{R}^\tau$—called a STRS induced from $\mathcal{R}$ and τ. Note that $\mathcal{R}^\tau$ acts on $\mathcal{T}^\tau$, i.e. $s, t \in \mathcal{T}^\tau$ whenever $s \to_{\mathcal{R}^\tau} t$; and that for any $s, t \in \mathcal{T}^\tau$ $s \to_{\mathcal{R}} t$ if and only if $s \to_{\mathcal{R}^\tau} t$.

Using the notion of sort attachments, persistency can now be alternatively formulated as follows.

Proposition 2. *A property $\mathcal{P}$ of TRSs is persistent if for any TRS $\mathcal{R}$ and for any attachment τ consistent with $\mathcal{R}$,*

$$\mathcal{R}^\tau \textit{ has property } \mathcal{P} \Leftrightarrow \mathcal{R} \textit{ has property } \mathcal{P}.$$

We will prove the persistency of confluence in the form of Proposition 2. From now on, we assume that a set $\mathcal{S}$ of sort and a TRS $\mathcal{R}$ are given and that an attachment τ on $\mathcal{S}$ that is consistent with $\mathcal{R}$ is fixed. Then our goal is to show $\mathcal{R}^\tau$ is confluent if and only if $\mathcal{R}$ is confluent. The proof of the (if)-part presents no difficulties, and therefore we concentrate on the (only if)-part; we assume the confluence of well-sorted terms and prove the confluence of unsorted terms. In the sequel, unsorted terms are often referred to as just terms.

3 Persistency of confluence

3.1 Characterizations by well-sortedness

Definition 3. 1. The *top sort* of a term $t \in \mathcal{T}$ is defined by

$$\mathrm{top}(t) = \begin{cases} \tau(t) & \text{if } t \in \mathcal{V}, \\ \beta & \text{if } t \equiv f(t_1, \ldots, t_n) \text{ with } f : \alpha_1 \times \cdots \times \alpha_n \to \beta. \end{cases}$$

2. Let $t \equiv C[t_1, \ldots, t_n] \in \mathcal{T}$ $(n \geq 0)$ be a term with $C \not\equiv \Box$. We write $t \equiv C[\![t_1, \ldots, t_n]\!]$ if (1) $C : \alpha_1 \times \cdots \times \alpha_n \to \beta$ and (2) $\mathrm{top}(t_i) \neq \alpha_i$ for $i = 1, \ldots, n$. If this is the case, the terms $t_1, \ldots, t_n$ are called *principal subterms* of t. Clearly, a term t is uniquely written as $C[\![t_1, \ldots, t_n]\!]$ for some $C \in \mathcal{C}$ and terms $t_1, \ldots, t_n$.

Definition 4. A rewrite step $s \to t$ is said to be *inner* (written as $s \to^i t$) if

$$s \equiv C[\![s_1, \ldots, C'[l\sigma], \ldots, s_n]\!] \to C[s_1, \ldots, C'[r\sigma], \ldots, s_n] \equiv t$$

for some terms $s_1, \ldots, s_n$, a substitution σ, $l \to r \in \mathcal{R}$, and $C' \in \mathcal{C}$; otherwise it is *outer* (written as $s \to^o t$).

Definition 5. A rewrite step $s \to^o t$ is said to be *destructive at level* 1 if $\mathrm{top}(s) \neq \mathrm{top}(t)$. The rewrite step $s \to t$ is said to be *destructive at level* $k+1$ if $s \equiv C[\![s_1, \ldots, s_j, \ldots, s_n]\!] \to^i C[s_1, \ldots, t_j, \ldots, s_n] \equiv t$ with $s_j \to t_j$ destructive at level k.

Lemma 6. *A rewrite step $s \to^o t$ is destructive at level 1 if and only if $s \equiv C[\![s_1, \ldots, \sigma(x), \ldots, s_n]\!]$ and $\sigma(x) \equiv t$ for some terms $s_1, \ldots, s_n$, a substitution σ, and $C \in \mathcal{C}$ such that $C[s_1, \ldots, \Box, \ldots, s_n] \equiv C'\sigma$ for some $C'[x] \to x \in \mathcal{R}$.*

Proof. ($\Leftarrow$) Suppose $\mathrm{top}(s) = \mathrm{top}(t)$. Then, since $C' \not\equiv \Box$ by the definition of rewrite rules, $\mathrm{top}(C'[x]) = \mathrm{top}(C'[x]\sigma) = \mathrm{top}(\sigma(x))$. Also, by consistency, $\mathrm{top}(x) = \mathrm{top}(C'[x])$, and so $\mathrm{top}(x) = \mathrm{top}(t)$. But then, t can not be principal, since $C[s_1, \ldots, \Box, \ldots, s_n] \equiv C'\sigma : \mathrm{top}(x) \to \mathrm{top}(s)$. ($\Rightarrow$) Suppose $\mathrm{top}(s) \neq \mathrm{top}(t)$. By consistency, the rewrite step $s \to_{\mathcal{R}} t$ is an application of a collapsing rule, and the redex of the rewrite step is s. Let the rule be $C'[x] \to x$, and suppose $s \equiv C'[x]\sigma$ and $t \equiv \sigma(x)$. Since $C[s_1, \ldots, \Box, \ldots, s_n] : \mathrm{top}(x) \to \mathrm{top}(s)$, it suffices to show $\mathrm{top}(x) \neq \mathrm{top}(t)$. But $\mathrm{top}(x) = \mathrm{top}(C'[x])$ by consistency, and $\mathrm{top}(C'[x]) = \mathrm{top}(C'[x]\sigma) = \mathrm{top}(s)$ since $C' \not\equiv \Box$. Hence $\mathrm{top}(x) = \mathrm{top}(s) \neq \mathrm{top}(t)$. $\Box$

This proposition shows that a destructive rewrite step occurs only when the applied rule is collapsing, and that the reduct of a destructive rewrite step indeed results from one of the principal subterms.

The next lemma, which will be often used implicitly in the sequel, is proved in a straightforward way; it analyzes the structure of a rewrite step. We write $t \equiv C\langle\!\langle t_1, \ldots, t_n \rangle\!\rangle$ when either $t \equiv C[\![t_1, \ldots, t_n]\!]$ or $C \equiv \Box$ and $t \equiv t_1$.

Lemma 7. *1. If $s \to^o t$ then*

$$\begin{cases} s \equiv C[\![s_1, \ldots, s_n]\!], \\ t \equiv C^*\langle\!\langle s_{i_1}, \ldots, s_{i_m} \rangle\!\rangle \quad \text{where } i_1, \ldots, i_m \in \{1, \ldots, n\} \end{cases}$$

for some $C, C^* \in \mathcal{C}$ and terms $s_1, \ldots, s_n$ and either

(a) $n = m = 0$ and s and t are well-sorted;

(b) $n \neq 0$ and $s \to^o t$ is destructive (at level 1) and $C^* \equiv \Box$ and $t \equiv s_j$ for some $1 \leq j \leq n$; or

(c) $n \neq 0$ and $s \to^o t$ is not destructive (at level 1) and $t \equiv C^*[\![s_{i_1}, \ldots, s_{i_m}]\!]$.

2. If $s \to^i t$ then

$$\begin{cases} s \equiv C[\![s_1, \ldots, s_j, \ldots, s_n]\!], \\ t \equiv C[s_1, \ldots, t_j, \ldots, s_n], \\ s_j \to t_j \end{cases}$$

for some $C \in \mathcal{C}$ and terms $s_1, \ldots, s_n, t_j$ and either

(a) $s_j \to t_j$ is destructive at level 1, t_j is a principal subterm of s_j and either

i. $t \equiv C[\![s_1, \ldots, t_j, \ldots, s_n]\!]$, or

ii. $t_j \equiv C'[\![u_1, \ldots, u_l]\!]$ and $t \equiv C''[\![s_1, \ldots, u_1, \ldots, u_l, \ldots, s_n]\!]$ for some $C', C'' \in \mathcal{C}$ and terms $u_1, \ldots, u_l$; or

(b) $s_j \to t_j$ is not destructive at level 1 and $t \equiv C[\![s_1, \ldots, t_j, \ldots, s_n]\!]$.

Definition 8. The *rank* of a term t is defined by

$$\mathrm{rank}(t) = \begin{cases} 1 & \text{if } t \text{ is well-sorted,} \\ 1 + \max\{\mathrm{rank}(t_i) \mid 1 \leq i \leq n\} & \text{if } t \equiv C[\![t_1, \dots, t_n]\!] \text{ with } n \geq 1. \end{cases}$$

Lemma 9. *1. If* $s \to t$ *then* $\mathrm{rank}(s) \geq \mathrm{rank}(t)$.
2. If a rewrite step $s \to t$ *is destructive at level* 1 *then* $\mathrm{rank}(s) > \mathrm{rank}(t)$.

Proof. 1. Our proof proceeds by induction on the rank of s. Base step is trivial. Suppose $\mathrm{rank}(s) = n(> 1)$. We distinguish two cases:

(a) $s \to^o t$. If $s \to^o t$ is destructive, then $s \equiv C[\![s_1, \dots, s_n]\!] \to s_j \equiv t$, and so $\mathrm{rank}(s) > \mathrm{rank}(t)$ by the definition of rank. Otherwise $s \equiv C[\![s_1, \dots, s_n]\!] \to^o t \equiv C^*[\![s_{i_1}, \dots, s_{i_m}]\!]$. Then,

$$\begin{aligned} \mathrm{rank}(s) &= \max\{\mathrm{rank}(s_1), \dots, \mathrm{rank}(s_n)\} + 1 \\ &\geq \max\{\mathrm{rank}(s_{i_1}), \dots, \mathrm{rank}(s_{i_m})\} + 1 \\ &= \mathrm{rank}(t). \end{aligned}$$

(b) $s \to^i t$. Then $s \equiv C[\![s_1, \dots, s_j, \dots, s_n]\!]$, $t \equiv C[s_1, \dots, t_j, \dots, s_n]$ and $s_j \to t_j$ for some $C \in \mathcal{C}$ and terms $s_1, \dots, s_n, t_j$. We distinguish two cases:

i. $t \equiv C[\![s_1, \dots, t_j, \dots, s_n]\!]$. Then, using the induction hypothesis,

$$\begin{aligned} \mathrm{rank}(s) &= \max\{\mathrm{rank}(s_1), \dots, \mathrm{rank}(s_j), \dots, \mathrm{rank}(s_n)\} + 1 \\ &\geq \max\{\mathrm{rank}(s_1), \dots, \mathrm{rank}(t_j), \dots, \mathrm{rank}(s_n)\} + 1 \\ &= \mathrm{rank}(t). \end{aligned}$$

ii. $t_j \equiv C'[\![u_1, \dots, u_l]\!]$ and $t \equiv C''[\![s_1, \dots, u_1, \dots, u_l, \dots, s_n]\!]$ for some $C', C'' \in \mathcal{C}$ and terms $u_1, \dots, u_l$. By definition, $\mathrm{rank}(t_j) > \mathrm{rank}(u_i)$ for all $i = 1, \dots, l$. Therefore, using the induction hypothesis,

$$\begin{aligned} &\mathrm{rank}(s) \\ &= \max\{\mathrm{rank}(s_1), \dots, \mathrm{rank}(s_j), \dots, \mathrm{rank}(s_n)\} + 1 \\ &\geq \max\{\mathrm{rank}(s_1), \dots, \mathrm{rank}(t_j), \dots, \mathrm{rank}(s_n)\} + 1 \\ &\geq \max\{\mathrm{rank}(s_1), \dots, \mathrm{rank}(u_1), \dots, \mathrm{rank}(u_l), \dots, \mathrm{rank}(s_n)\} + 1 \\ &= \mathrm{rank}(t). \end{aligned}$$

2. Since $s \to t$ is destructive at level 1, $s \equiv C[\![s_1, \dots, s_n]\!] \to s_j \equiv t$, and so $\mathrm{rank}(s) > \mathrm{rank}(t)$ by the definition of rank.

□

3.2 Existence of preserved reducts

Definition 10. A term t is said to be a *level* 1 *special subterm* of t. If $t \equiv C[\![t_1, \dots, t_n]\!]$ then level k special subterms of $t_1, \dots, t_n$ are said to be *level* $k+1$ *special subterms* of t. The multiset of all level k special subterms of a term t is denoted by $S_k(t)$. Moreover $\sum_{k \geq 1} S_k(t)$ is denoted by $S(t)$. Here $\sum$ denotes the multiset sum.

Note that $S_2(t)$ denotes the multiset of principal terms of t.

Lemma 11. *Let $s \to t$ be a rewrite step destructive at level n. Then there exist a level n special subterm s' of s such that $s \equiv C[s']$ and a term t' such that $t \equiv C[t']$ and $s' \to t'$ destructive at level 1.*

Proof. Our proof proceeds by induction on the level of destructive rewrite step. Base step is trivial. For the induction step, suppose that $s \to t$ is a rewrite step destructive at level $k+1$, i.e. $s \equiv C[\![s_1, \ldots, s_j, \ldots, s_n]\!] \to C[s_1, \ldots, t_j, \ldots, s_n] \equiv t$ with $s' \to t'$ destructive at level k. Then, by induction hypothesis, there exists a level k special subterm s' of s_j—hence, s' is a level $k+1$ special subterm of s—such that $s_j \equiv C'[s']$ and a term t' such that $t_j \equiv C'[t']$ and $s' \to t'$ destructive at level 1. □

Lemma 12. *Let $s \to t$ be a rewrite step. If $t' \in S(t)$ then there exists $s' \in S(s)$ such that $s' \stackrel{\equiv}{\to} t'$.*

Proof. Our proof proceeds by induction on the level of a special subterm t' in t. Base step is trivial. For the induction step, suppose $t' \in S_{k+1}(t)$. We distinguish two cases:

1. $s \to^o t$. Then $s \equiv C[\![s_1, \ldots, s_n]\!]$ and $t \equiv C^* \langle\!\langle s_{i_1}, \ldots, s_{i_m} \rangle\!\rangle$ with $i_1, \ldots, i_m \in \{1, \ldots, n\}$ for some $C, C^* \in \mathcal{C}$ and terms $s_1, \ldots, s_n$. Therefore $t' \in S(s_j) \subseteq S(s)$, and so we can put $s' \equiv t'$.
2. $s \to^i t$. Then $s \equiv C[\![s_1, \ldots, s_j, \ldots, s_n]\!]$, $t \equiv C[s_1, \ldots, t_j, \ldots, s_n]$ and $s_j \to t_j$ for some $C \in \mathcal{C}$ and terms $s_1, \ldots, s_n, t_j$. We distinguish two cases:
   (a) $t \equiv C[\![s_1, \ldots, t_j, \ldots, s_n]\!]$. If $t' \in S_k(s_i)$ then, since $S(s_i) \subseteq S(s)$, we can put $s' \equiv t'$. Otherwise $t' \in S_k(t_j)$. Then, since $s_j \to t_j$ and $t' \in S_k(t_j)$, there exists $s' \in S(s_j) \subseteq S(s)$ such that $s' \stackrel{\equiv}{\to} t'$ by induction hypothesis.
   (b) $t_j \equiv C'[\![u_1, \ldots, u_l]\!]$ and $t \equiv C''[\![s_1, \ldots, u_1, \ldots, u_l, \ldots, s_n]\!]$ for some C', $C'' \in \mathcal{C}$ and terms $u_1, \ldots, u_l$. If $t' \in S_k(s_i)$ then, since $S(s_i) \subseteq S(s)$, we can put $s' \equiv t'$. Otherwise $t' \in S_k(u_i)$. Then, since $s_j \to^o t_j$, we have $S(u_i) \subseteq S(s_j)$, and so $t' \in S(s_j) \subseteq S(s)$.

□

Lemma 13. *Let $s \equiv C[s'] \to C[t'] \equiv t$ be a rewrite step with $s' \to t'$ and $s' \in S(s)$. If $s' \to t'$ is not destructive at level 1 then $t' \in S(t)$.*

Proof. It immediately follows from the definition. □

Definition 14. A term s is said to be *preserved* if there exists no reduction sequence starting from s that contains a destructive rewrite step.

Definition 15. We define a *collapsing reduction* ($\to_c$, in symbol) as follows: $s \to_c t$ if there exist a special subterm s' of s such that $s \equiv C[s']$ and a term t' such that $s' \stackrel{*}{\to} t'$, $t \equiv C[t']$ and $s' \stackrel{*}{\to} t'$ contains a level 1 destructive rewrite step; if this is the case, the *collapsing redex* of this reduction is s'.

Lemma 16. *1. If $s \to_c t$ then $s \stackrel{*}{\to} t$.*
2. A term is preserved if and only if it contains no collapsing redex.

Proof. 1. Easy. 2. ($\Leftarrow$) Assume there exists a reduction sequence starting from s that contains a destructive rewrite step, namely $s \stackrel{*}{\to} t \to u$ with $t \to u$ destructive. Then, by Lemma 11, there exist a special subterm t' of t such that $t \equiv C'[t']$ and a term u' such that $u \equiv C'[u']$. and $t' \to u'$ is destructive at level 1. Then repeated applications of Lemma 12 yield $s' \in S(s)$ with $s' \stackrel{*}{\to} t' \to u'$; this s' is a collapsing redex in s. ($\Rightarrow$) Suppose $s \to_c t$. Then, by definition, there exists a special subterm s' of s such that $s \equiv C[s']$ and a term t' such that $s' \stackrel{*}{\to} t'$, $t \equiv C[t']$ and $s' \stackrel{*}{\to} t'$ contains a level 1 destructive rewrite step. Let the first level 1 destructive step in $s' \stackrel{*}{\to} t'$ be $u' \to v'$. Then, since $s' \in S(s)$, we have $u' \in S(C[u'])$ by Lemma 13, and so $C[u'] \to C[v']$ is destructive. Hence $s \equiv C[s'] \stackrel{*}{\to} C[u'] \to C[v'] \stackrel{*}{\to} C[t'] \equiv t$ contains a destructive rewrite step. $\square$

Lemma 17. *Every term has a preserved reduct.*

Proof. We show that there exists no infinite collapsing reduction sequence. Then the proposition immediately follows from Lemma 16.

For any term t, we assign a multiset $\|t\|$ by $\|t\| = [\mathrm{rank}(s) \mid s \in S(t)]$. We are now going to show that $s \to_c t$ implies $\|s\| \gg \|t\|$. Here $\gg$ is the multiset extension of the standard ordering $>$ on natural numbers.

Suppose $s \to_c t$. Then, by definition, there exist a special subterm s' of s such that $s \equiv C[s']$ and a term t' such that $s' \stackrel{*}{\to} t'$, $t \equiv C[t']$ and $s' \stackrel{*}{\to} t'$ contains a level 1 destructive rewrite step. Then $\mathrm{rank}(s') > \mathrm{rank}(t')$ by Lemma 9, and therefore $\|s\| \gg \|t\|$, because s' is a special subterm of s. Now, since the relation $\gg$ is well-founded (see e.g. [Dershowitz 1979]), this shows that there exists no infinite collapsing reduction sequence. $\square$

3.3 Confluence of inner preserved terms

Let $s_1, \ldots, s_n$ and $t_1, \ldots, t_n$ be terms. We write $\langle s_1, \ldots, s_n\rangle \propto \langle t_1, \ldots, t_n\rangle$ if for any $1 \le i, j \le n$, $s_i \equiv s_j$ implies $t_i \equiv t_j$ holds; and $\langle s_1, \ldots, s_n\rangle \infty \langle t_1, \ldots, t_n\rangle$ if both $\langle s_1, \ldots, s_n\rangle \propto \langle t_1, \ldots, t_n\rangle$ and $\langle t_1, \ldots, t_n\rangle \propto \langle s_1, \ldots, s_n\rangle$ hold. Then the following lemma is proved in a straightforward way.

Lemma 18. *If* $C[\![s_1, \ldots, s_n]\!] \to^o C'\langle\!\langle s_{i_1}, \ldots, s_{i_m}\rangle\!\rangle$ *and* $\langle s_1, \ldots, s_n\rangle \propto \langle t_1, \ldots, t_n\rangle$ *then* $C[t_1, \ldots, t_n] \to^o C'[t_{i_1}, \ldots, t_{i_m}]$.

Lemma 19. *If rewrite steps* $t_1 \stackrel{*}{\leftarrow}{}^o\, t \stackrel{*}{\to}{}^o\, t_2$ *do not contain a destructive rewrite step except the last steps then* t_1 *and* t_2 *are joinable.*

Proof. By Lemma 7, we may write $t \equiv C[\![s_1, \ldots, s_n]\!]$, $t_1 \equiv C_1\langle\!\langle s_{i_1}, \ldots, s_{i_m}\rangle\!\rangle$ and $t_2 \equiv C_2\langle\!\langle s_{j_1}, \ldots, s_{j_p}\rangle\!\rangle$. Let $C : \alpha_1 \times \cdots \times \alpha_n \to \beta$. Choose fresh variables $x_i \in \mathcal{V}_{\alpha_i}$ for $i = 1, \ldots, n$ such that $\langle x_1, \ldots, x_n\rangle \infty \langle s_1, \ldots, s_n\rangle$. Let $t' \equiv C[x_1, \ldots, x_n]$, $t'_1 \equiv C_1[x_{i_1}, \ldots, x_{i_m}]$ and $t'_2 \equiv C_2[x_{j_1}, \ldots, x_{j_p}]$. Then repeated applications of Lemma 18 yield well-sorted rewrite steps $t'_1 \stackrel{*}{\leftarrow} t' \stackrel{*}{\to} t'_2$. Therefore, by our assumption that well-sorted terms are confluent, t'_1 and t'_2 are joinable, i.e. $t'_1 \stackrel{*}{\to} C^*[x_{k_1}, \ldots, x_{k_q}] \stackrel{*}{\leftarrow} t'_2$ for some C^*. Instantiating this rewrite steps, we have $t_1 \stackrel{*}{\to}{}^o\, C^*\langle\!\langle s_{k_1}, \ldots, s_{k_q}\rangle\!\rangle \stackrel{*}{\leftarrow}{}^o\, t_2$. $\square$

Definition 20. Let S be a set of confluent terms. A set $\widehat{S}$ of terms *represents* S if the following two conditions are satisfied:

1. Every term s in S has a unique reduct $\widehat{s}$ in $\widehat{S}$, which will be called the *representative of* s.
2. Joinable terms in S have the same representative in $\widehat{S}$.

Lemma 21. *Every finite set S of confluent terms can be represented.*

Proof. Since S consists of confluent terms, joinability is an equivalence relation on S. Since S is finite, each equivalence class is also finite and has a common reduct. It is obvious that the set of such common reducts represents S. $\square$

Definition 22. A term s is said to be *inner preserved* if its all principal subterms are preserved.

Lemma 23. *Inner preserved terms are confluent.*

Proof. We show that every inner preserved term t is confluent by induction on the rank of t. Base step is trivial, by our assumption that well-sorted terms are confluent. For the induction step, suppose that $t \equiv C[\![s_1, \ldots, s_n]\!]$ is an inner preserved term with rank $n(> 1)$ and $t_1 \overset{*}{\leftarrow} t \overset{*}{\rightarrow} t_2$.

First, note that a destructive rewrite step occurs in $t \overset{*}{\rightarrow} t_i \quad (i = 1, 2)$ only once and is level 1, because of our assumption that t is inner preserved.

For $i = 1, 2$, we are now going to define sets $\mathcal{L}(t \overset{*}{\rightarrow} t_i)$. Let $t \overset{*}{\rightarrow} t_i$ be $t \equiv w_0 \rightarrow w_1 \rightarrow \cdots \rightarrow w_l \equiv t_i$, and suppose $w_k \rightarrow w_{k+1}$ is destructive (put $k = l$ if there is no destructive rewrite step). Then let

$$\mathcal{L}(t \overset{*}{\rightarrow} t_i) = \{u \mid u \in S_2(w_j), 0 \leq j \leq k\} \cup \{w_j \mid k < j \leq l\}.$$

Let S be $\mathcal{L}(t \overset{*}{\rightarrow} t_1) \cup \mathcal{L}(t \overset{*}{\rightarrow} t_2)$. Then for any elements $s \in S$, (1) s is also inner preserved, because s is preserved and (2) s has less rank than n by the definition of rank and Lemma 9. Therefore we can apply the induction hypothesis to each element of S, and so S is a set of confluent terms. Hence, by Lemma 21, S has a representation $\widehat{S}$.

Next we define $\tilde{u}$ for each term u that appears in $t_1 \overset{*}{\leftarrow} t \overset{*}{\rightarrow} t_2$ as follows. If u is a term that appears before the destructive rewrite step occurs then define $\tilde{u}$ to be a term obtained from u by replacing all its principal subterms with its representative. Otherwise define $\tilde{u}$ to be the representative of u. Note that $u \overset{*}{\rightarrow} \tilde{u}$ in whichever case.

Let $u_1 \rightarrow u_2$ be a a rewrite step in $t_1 \overset{*}{\leftarrow} t \overset{*}{\rightarrow} t_2$. We are now going to show that 1. $\tilde{u_1} \overset{\equiv}{\rightarrow}{}^o \tilde{u_2}$ if $u_1 \rightarrow u_2$ is not preceded by the destructive rewrite step; and 2. $\tilde{u_1} \equiv \tilde{u_2}$ if $u_1 \rightarrow u_2$ is preceded by the destructive rewrite step.

1. We distinguish two cases:
   (a) $u_1 \rightarrow^o u_2$. Then $u_1 \equiv C[\![s_1, \ldots, s_n]\!]$ and $u_2 \equiv C'\langle\!\langle s_{i_1}, \ldots, s_{i_m}\rangle\!\rangle$. By definition, $\tilde{u_1} \equiv C[\![\widehat{s_1}, \ldots, \widehat{s_n}]\!]$ and $\tilde{u_2} \equiv C'\langle\!\langle \widehat{s_{i_1}}, \ldots, \widehat{s_{i_m}}\rangle\!\rangle$. Since $\langle s_1, \ldots, s_n\rangle \propto \langle \widehat{s_1}, \ldots, \widehat{s_n}\rangle$ by the definition of representation, we conclude $\tilde{u_1} \rightarrow^o \tilde{u_2}$ by Lemma 18.

(b) $u_1 \rightarrow^i u_2$. Then, since t is inner preserved, $u_1 \equiv C[\![s_1, \ldots, s_j, \ldots, s_n]\!]$, $u_2 \equiv C[\![s_1, \ldots, s'_j, \ldots, s_n]\!]$ with $s_j \rightarrow s'_j$. Then, since $\widehat{s_j} \equiv \widehat{s'_j}$ by the definition of representation, we have $\tilde{u_1} \equiv \tilde{u_2}$.

2. We have $u_1, u_2 \in S$ and $\widehat{u_1} \equiv \widehat{u_2}$ by the definition of representation; hence $\tilde{u_1} \equiv \tilde{u_2}$.

From this, we conclude that $\tilde{t_1} \overset{*}{\leftarrow}{}^{o} \tilde{t} \overset{*}{\rightarrow}{}^{o} \tilde{t_2}$, and moreover that $\tilde{t_1} \overset{*}{\leftarrow}{}^{o} \tilde{t} \overset{*}{\rightarrow}{}^{o} \tilde{t_2}$ do not contain a destructive rewrite step except the last steps. Therefore $\tilde{t_1}$ and $\tilde{t_2}$ are joinable by Lemma 19; hence so are t_1 and t_2 [see Figure 1]. □

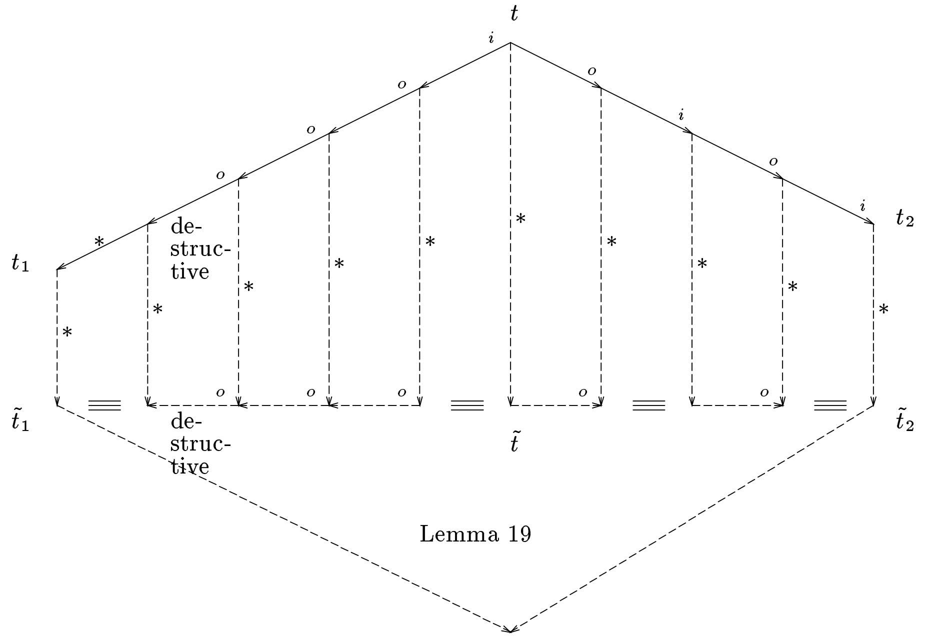

Figure 1: Lemma 23

3.4 Persistency of confluence

Definition 24. Let $s \equiv C[\![s_1, \ldots, s_n]\!]$ be a term. Then a term $t \equiv C[t_1, \ldots, t_n]$ which satisfies (1) t is inner preserved (2) $s_i \overset{*}{\rightarrow} t_i$ for $i = 1, \ldots, n$ and (3) $\langle s_1, \ldots, s_n \rangle \propto \langle t_1, \ldots, t_n \rangle$, is called a *witness* of s.

Lemma 25. *Every term has a witness.*

Proof. Let $s \equiv C[\![s_1, \ldots, s_n]\!]$. By Lemma 17, there exists a preserved reduct t_i of s_i for each $i = 1, \ldots, n$. Taking the same preserved term for identical terms ensures $\langle s_1, \ldots, s_n \rangle \propto \langle t_1, \ldots, t_n \rangle$. Note that every principal term of $t \equiv C[t_1, \ldots, t_n]$ is under or equal to one of t_i's. Therefore, since t_i's are all preserved, so are all the principal terms of t. Thus t is a witness of s. $\square$

In the sequel, an arbitrary witness of a term s is denoted by $\dot{s}$.

$$\begin{tikzcd}
s \equiv C\llbracket s_1, \ldots, s_n \rrbracket \arrow[rr, "o"] \arrow[d, "*"] & & C^*\langle\langle s_{i_1}, \ldots, s_{i_m} \rangle\rangle \equiv t \arrow[dd, "*"] \\
\dot{s} \equiv C[u_1, \ldots, u_n] \arrow[d, dashed] & & \\
C^*[u_{i_1}, \ldots, u_{i_m}] \arrow[dr, dashed, "*"] & & C^*[t_{i_1}, \ldots, t_{i_m}] \equiv \dot{t} \arrow[dl, dashed, "*"'] \\
& {} &
\end{tikzcd}$$

Figure 2: Lemma 26

Lemma 26. *Let s, t be terms such that $s \to t$. If all the principal subterms of s are confluent then $\dot{s}$ and $\dot{t}$ are joinable.*

Proof. Let $s \equiv C\llbracket s_1, \ldots, s_n \rrbracket$, and suppose that $s_1, \ldots, s_n$ are confluent, and that $\dot{s} \equiv C[u_1, \ldots, u_n]$ for respective reducts $u_1, \ldots, u_n$ of $s_1, \ldots, s_n$. We distinguish two cases:

1. $s \to^o t$. Then we have $t \equiv C^*\langle\langle s_{i_1}, \ldots, s_{i_m} \rangle\rangle$ for some $C^* \in \mathcal{C}$. Suppose $\dot{t} \equiv C^*[t_{i_1}, \ldots, t_{i_m}]$ for respective reducts $t_{i_1}, \ldots, t_{i_m}$ of $s_{i_1}, \ldots, s_{i_m}$. Since $\langle s_1, \ldots, s_n \rangle \propto \langle u_1, \ldots, u_n \rangle$, we have $\dot{s} \equiv C[u_1, \ldots, u_n] \to C^*[u_{i_1}, \ldots, u_{i_m}]$ by Lemma 18. Now u_j and t_j are joinable for each $j = i_1, \ldots, i_m$, because $u_j \stackrel{*}{\leftarrow} s_j \stackrel{*}{\to} t_j$. Therefore $\dot{s}$ and $\dot{t}$ are joinable [see Figure 2].
2. $s \to^i t$. Then $t \equiv C[s_1, \ldots, s'_j, \ldots, s_n]$ and $s_j \to s'_j$. It is easy to see $\dot{t} \equiv C[t_1, \ldots, t_n]$ for some respective reducts $t_1, \ldots, t_n$ of $s_1, \ldots, s'_j, \ldots, s_n$. Now u_j and t_j are joinable for each $j = i_1, \ldots, i_m$, because $u_j \stackrel{*}{\leftarrow} s_j \stackrel{*}{\to} t_j$. Therefore $\dot{s}$ and $\dot{t}$ are joinable.

$\square$

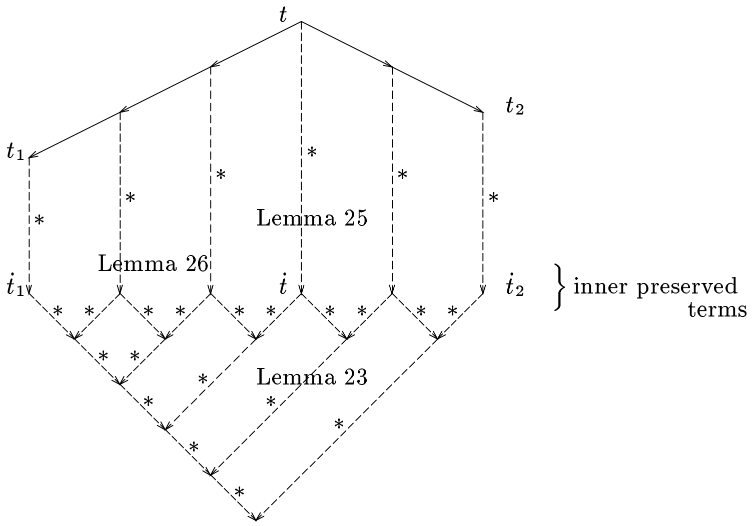

Figure 3: Theorem 27

Theorem 27. *Confluence is a persistency property of TRSs.*

Proof. Suppose that every well-sorted term is confluent. We show that every term t is confluent by induction on the rank of t. Base step is trivial. Suppose $\mathrm{rank}(t) = n(> 1)$ and $t_1 \overset{*}{\leftarrow} t \overset{*}{\rightarrow} t_2$. Note that every term that appears in $t_1 \overset{*}{\leftarrow} t \overset{*}{\rightarrow} t_2$ has the rank less than or equal to n by Lemma 9; therefore its principal subterms have the ranks less than n, and hence they are confluent by induction hypothesis. Now, by Lemma 25, any term that appears in $t_1 \overset{*}{\leftarrow} t \overset{*}{\rightarrow} t_2$ has a witness. Then repeated applications of Lemma 26 yield a conversion between $\dot{t}_1$ and $\dot{t}_2$ via witnesses. Since witnesses are inner preserved, we can apply Lemma 23 repeatedly so that $\dot{t}_1$ and $\dot{t}_2$ are joinable. Hence so are t_1 and t_2 [see Figure 3]. $\square$

Example 1. Let

$$\mathcal{R} \begin{cases} f(x,y) \rightarrow f(g(x), g(x)) & (r1) \\ g(x) \rightarrow h(x) & (r2) \\ F(g(x), x) \rightarrow F(x, g(x)) & (r3) \\ F(h(x), x) \rightarrow F(x, h(x)). & (r4) \end{cases}$$

It is difficult to show the confluence of $\mathcal{R}$ directly, because $\mathcal{R}$ is neither terminating nor orthogonal. Also, we can not use the direct sum result because of function symbols g and h.

Think of attachment τ such that $f : 0 \times 0 \rightarrow 1$, $g : 0 \rightarrow 0$, $h : 0 \rightarrow 0$ and $F : 0 \times 0 \rightarrow 2$. It is easy to check τ is consistent with $\mathcal{R}$. We are now going to show $\mathcal{R}^\tau$ is confluent i.e. every well-sorted term t is confluent. We distinguish three cases:

– $t \in \mathcal{T}^0$. Since $(r2)$ is the only applicable rule, t is confluent.

- $t \in \mathcal{T}^1$. Then $(r1)$ and $(r2)$ are the only rules applicable to t. Since a TRS $\{(r1),(r2)\}$ is orthogonal, it is confluent. Therefore t is confluent.
- $t \in \mathcal{T}^2$. Then $(r2)$, $(r3)$ and $(r4)$ are the only rules applicable to t. Since a TRS $\{(r2),(r3),(r4)\}$ is terminating and its critical pair is joinable, it is confluent. Therefore t is confluent.

Now the confluence of $\mathcal{R}$ follows from that of $\mathcal{R}^\tau$ by the persistency of confluence.

Acknowledgments

This work is partially supported by Grants from Ministry of Education, Science and Culture of Japan, #09245212 and #07680347.

References

[Aoto and Toyama 1996] Aoto, T. and Toyama, Y.: "Persistency of confluence"; Research Report IS-RR-96-0009F, School of Information Science, JAIST (1996).

[Aoto and Toyama 1997] Aoto, T. and Toyama, Y.: "On composable properties of term rewriting systems"; Proceedings of the 6th International Conference on Algebraic and Logic Programming, LNCS 1298, Springer (1997), 114–128.

[Dershowitz 1979] Dershowitz, N. and Manna, Z.: "Proving termination with multiset orderings"; Communications of the ACM 22, 8 (1979) 465–476.

[Gramlich 1995] Gramlich, B.: "Abstract relations between restricted termination and confluence properties of rewrite systems"; Fundamenta Informaticae 24 (1995), 3–23.

[Huet 1980] Huet, G.: "Confluent reductions: abstract properties and applications to term rewriting systems"; Journal of the Association for Computing Machinery 27, 4 (1980), 797–821.

[Kitahara 1995] Kitahara, A.: "On the modularity of confluent term rewriting systems with shared constructors (in Japanese)"; Master's thesis, JAIST (1995).

[Klop et al. 1994] Klop, J. W., Middeldorp, A., Toyama, Y. and de Vrijer, R.: "Modularity of confluence: A simplified proof"; Information Processing Letters 49 (1994), 101–109.

[Knuth and Bendix 1970] Knuth, D. E. and Bendix, P. B.: "Simple word problems in universal algebras"; Computational Problems in Abstract Algebra (Leech, J., ed.), Pergamon Press (1970), 263–297.

[Marchiori 1996] Marchiori, M.: "On the modularity of normal forms in rewriting"; Journal of Symbolic Computation 22 (1996), 143–154.

[Middeldorp 1989a] Middeldorp, A.: "Modular aspects of properties of term rewriting systems related to normal forms"; Proceedings of the 3rd International Conference on Rewriting Techniques and Applications, LNCS 355, Springer (1989), 263–277.

[Middeldorp 1989b] Middeldorp, A.: "A sufficient condition for the termination of the direct sum of term rewriting systems"; Proceedings of the 4th annual IEEE Symposium on Logic in Computer Science (1989), 396–401.

[Middeldorp 1990] Middeldorp, A.: "Modular properties of term rewriting systems"; PhD thesis, Vrije Universiteit, Amsterdam (1990).

[Ohlebusch 1995] Ohlebusch, E.: "Modular properties of composable term rewriting systems"; Journal of Symbolic Computation 20, 1 (1995), 1–42.

[Rusinowitch 1987] Rusinowitch, M.: "On termination of the direct sum of term-rewriting systems"; Information Processing Letters 26 (1987), 65–70.

[Schmidt et al. 1995] Schmidt-Schauß, M., Marchiori, M. and Panitz, S. E.: "Modular termination of r-consistent and left-linear term rewriting systems"; Theoretical Computer Science 149 (1995), 361–374.

[Toyama 1987] Toyama, Y.: "On the Church-Rosser property for the direct sum of term rewriting systems"; Journal of the Association for Computing Machinery 34, 1 (1987), 128–143.

[Toyama 1988] Toyama, Y.: "Commutativity of term rewriting systems"; Programming of Future Generation Computers II (Fuchi, K. and Kott, L., eds.), North-Holland (1988), 393–407.

[Toyama et al. 1995] Toyama, Y., Klop, J. W. and Barendregt, H. P.: "Termination for direct sums of left-linear complete term rewriting systems"; Journal of the Association for Computing Machinery 42, 6 (1995), 1275–1304.

[Ohsaki and Middeldorp 1997] Ohsaki, H. and Middeldorp, A.: "Type introduction for equational rewriting"; Proceedings of the 4th Symposium on Logical Foundations of Computer Science, LNCS 1234, Springer (1997), 283–293.

[van de Pol 1993] van de Pol, J.: "Modularity in many-sorted term rewriting systems"; Master's thesis, Utrecht University (1993).

[Zantema 1994] Zantema, H.: "Termination of term rewriting: interpretation and type elimination"; Journal of Symbolic Computation 17 (1994), 23–50.