

Persistence of Termination for Term Rewriting Systems with Ordered Sorts

Munehiro Iwami

Abstract—A property P is persistent if for any many-sorted term rewriting system \mathcal{R} , \mathcal{R} has the property P if and only if term rewriting system $\Theta(\mathcal{R})$, which results from \mathcal{R} by omitting its sort information, has the property P . Zantema showed that termination is persistent for term rewriting systems without collapsing or duplicating rules. In this paper, we show that the Zantema's result can be extended to term rewriting systems on ordered sorts, i.e., termination is persistent for term rewriting systems on ordered sorts without collapsing, decreasing or duplicating rules. Furthermore we give the example as application of this result. Also we obtain that completeness is persistent for this class of term rewriting systems.

Keywords: Theory of computing, Model-based reasoning, term rewriting system, termination

I. INTRODUCTION

Term rewriting systems (TRSs) can offer both flexible computing and effective reasoning with equations and have been widely used as a model of functional and logic programming languages and as a basis of theorem provers, symbolic computation, algebraic specification and verification [4].

A rewrite system is called *terminating (strongly normalizing)* if there is no infinite rewrite sequence. The notion of termination for rewrite systems corresponds to the existence of answers of computations. So termination is the fundamental notion of term rewriting systems as computation models [7]. It is well-known that termination is undecidable for term rewriting systems in general. However, several sufficient approaches for proving this property have been successfully developed in particular cases.

Zantema [23] introduced the notion of *persistence* as follows. A property P is persistent if for any many-sorted TRS \mathcal{R} , \mathcal{R} has the property P if and only if TRS $\Theta(\mathcal{R})$, which results from \mathcal{R} by omitting its sort information, has the property P . Usual many-sorted TRS was extended with ordered sorts by Aoto and Toyama [2]. And it was shown that the persistency of confluence [1] is preserved for this extension in [2]. Zantema [23] showed that termination is persistent for TRSs without collapsing or duplicating rules. Ohsaki and Middeldorp [20] studied the persistence of termination, acyclicity and non-loopingness on equational many-sorted TRSs. Aoto proved that the persistence of termination for TRSs in which all variables are of the same sort [3]. We showed that the persistence of termination for non-overlapping TRSs [11]. Also, we showed that the persistence of termination for locally confluent

overlay TRSs [12]. And we showed that the persistence of termination for right-linear overlay TRSs [13]. Furthermore we showed that the persistence of semi-completeness for TRSs [14].

In this paper, we show that the above Zantema's result is preserved for Aoto and Toyama's extension in the subclass of order sorted term rewriting systems.

This research was first appeared in [9] and studied in [10]. Furthermore, Ohsaki [21] studied the case of equational order-sorted TRSs. Their equational order-sorted TRSs [21] were based on ordered-sorted algebras in [8], [22]. However, our TRSs on ordered sorts are based on Aoto and Toyama [2]. For example, we consider the sorts $Zero$ and Nat . If $Zero \prec Nat$ then $A_{Zero} \subset A_{Nat}$ where A_{Zero} and A_{Nat} are order-sorted algebras in equational order-sorted TRSs [21]. However, in our TRSs on ordered sorts we do not consider order-sorted algebras. In our research, if $Zero \prec Nat$ then $\mathcal{T}^{Zero} \cap \mathcal{T}^{Nat} = \emptyset$ holds where \mathcal{T}^{Zero} and \mathcal{T}^{Nat} are set of terms with sort $Zero$ and Nat , respectively. So our research does not depend on order-sorted algebras.

In section 2, many-sorted TRS is formulated on ordered sorts. Then, the persistence of termination on ordered sorts is shown in section 3 and 4. The proof is a generalization of a simplified proof of modularity of termination [18]. Furthermore we give the example as application of this result. Also we obtain that completeness is persistent for term rewriting systems on ordered sorts.

II. PRELIMINARIES

We mainly follow basic definitions and basic lemmas in the literature [2].

A. Sorted Term Rewriting Systems

In this subsection, we introduce the basic notions of sorted term rewriting systems. Usual term rewriting systems [4] are considered as special cases of sorted term rewriting systems.

Let \mathcal{S} be a set of *sorts* and \mathcal{V} be a set of countably infinite *sorted variables*. We assume that \mathcal{S} is equipped with a well-founded partial ordering \succ . We write $b \succeq b'$ if and only if $b \succ b'$ or $b = b'$.

We assume there is a set \mathcal{V}^b of countably infinite variables of sort b for each sort $b \in \mathcal{S}$. Let \mathcal{F} be a set of sorted function symbols. We assume that each sorted function symbol $f \in \mathcal{F}$ is given with the sorts of its arguments and the sort of its value, all of which are included in \mathcal{S} . We write $f:b_1 \times \dots \times b_n \rightarrow b'$ if and only if f takes n arguments of sorts

Manuscript received November 3, 2004.

M. Iwami is with the Faculty of Science and Engineering, Shimane University, Matsue, Shimane, 690-8504, Japan (e-mail: munehiro@cis.shimane-u.ac.jp).

b_1, \dots, b_n respectively to a value of sort b' . Function symbols of 0 argument are constants.

The set $\mathcal{T}(\mathcal{F}, \mathcal{V}) = \cup_{b \in \mathcal{S}} \mathcal{T}(\mathcal{F}, \mathcal{V})^b$ of all sorted terms built from \mathcal{F} and \mathcal{V} is defined as follows: (1) $\mathcal{V}^b \subseteq \mathcal{T}(\mathcal{F}, \mathcal{V})^b$, (2) $f: b'_1 \times \dots \times b'_n \rightarrow b'$, $t_i \in \mathcal{T}(\mathcal{F}, \mathcal{V})^{b'_i}$ and $b_i \preceq b'_i$ ($i = 1, \dots, n$) then $f(t_1, \dots, t_n) \in \mathcal{T}(\mathcal{F}, \mathcal{V})^{b'}$. Here $\mathcal{T}(\mathcal{F}, \mathcal{V})^b$ denotes the set of all terms of sort b .

We define the set of all strict sorted terms if (2) is replaced by (2') if $f: b'_1 \times \dots \times b'_n \rightarrow b'$, $t_i \in \mathcal{T}(\mathcal{F}, \mathcal{V})^{b'_i}$, $b_i \preceq b'_i$ ($i = 1, \dots, n$) and $b_j = b'_j$ whenever $t_j \in \mathcal{V}$ then $f(t_1, \dots, t_n) \in \mathcal{T}(\mathcal{F}, \mathcal{V})^{b'}$. We write $t : b$ if t is of sort b . $\mathcal{V}(t)$ denotes the set of all variables that appear in t . $\mathcal{T}(\mathcal{F}, \mathcal{V})^b$ and $\mathcal{T}(\mathcal{F}, \mathcal{V})$ are abbreviated as \mathcal{T}^b and \mathcal{T} , respectively. Let \square^b be a special constant (hole) of sort b . Elements of $\mathcal{T}(\mathcal{F} \cup \{\square^b \mid b \in \mathcal{S}\}, \mathcal{V})$ are called contexts over $\mathcal{T}(\mathcal{F}, \mathcal{V})$. We write $C: b_1 \times \dots \times b_n \rightarrow b'$ if and only if the sort of context C is b' and it has n holes $\square^{b_1}, \dots, \square^{b_n}$. If $C: b_1 \times \dots \times b_n \rightarrow b'$ and $t_1: b_1, \dots, t_n: b_n$ with $b_i \preceq b'_i$ ($i = 1, \dots, n$) then $C[t_1, \dots, t_n]$ denotes the term obtained from C by replacing holes with t_1, \dots, t_n from left to right. A context that contains one hole is denoted by $C[\]$. A term t is said to be a subterm of s if and only if $s = C[t]$ for some context C . A substitution θ is a mapping from \mathcal{V} to \mathcal{T} such that $x \in \mathcal{V}^b$ implies $\theta(x) \in \mathcal{T}^b$. A substitution over terms is defined as homomorphic extension. $\theta(t)$ is usually written as $t\theta$. A sorted rewrite rule on \mathcal{T} is a pair $l \rightarrow r$ such that $l \notin \mathcal{V}$, $\mathcal{V}(r) \subseteq \mathcal{V}(l)$, sorted terms l and r are strict and if $l : b$ and $r : b'$ then $b \succeq b'$. A sorted term rewriting system (STRS, for short) is a pair $(\mathcal{F}, \mathcal{R})$ where \mathcal{F} is a set of sorted function symbols and \mathcal{R} is a set of sorted rewrite rules on $\mathcal{T}(\mathcal{F}, \mathcal{R})$. $(\mathcal{F}, \mathcal{R})$ is often abbreviated as \mathcal{R} and in that case \mathcal{F} is defined to be the set of function symbols that appear in \mathcal{R} .

Given a STRS \mathcal{R} , a sorted term s is reduced to a sorted term t ($s \rightarrow_{\mathcal{R}} t$, in symbol) if and only if $s = C[l\theta]$ and $t = C[r\theta]$ for some rewrite rule $l \rightarrow r \in \mathcal{R}$, context C and substitution θ . We call $s \rightarrow_{\mathcal{R}} t$ a rewrite step or reduction from s to t of \mathcal{R} . θ is called redex of this rewrite step.

The transitive reflexive closure of $\rightarrow_{\mathcal{R}}$ is denoted by $\rightarrow_{\mathcal{R}}^*$. Terms t_1 and t_2 are joinable if there exists some term t' such that $t_1 \rightarrow_{\mathcal{R}}^* t' \leftarrow_{\mathcal{R}}^* t_2$. A term t is confluent if for any terms t_1 and t_2 , t_1 and t_2 are joinable whenever $t_1 \leftarrow_{\mathcal{R}}^* t \rightarrow_{\mathcal{R}}^* t_2$. A STRS \mathcal{R} is confluent if every term is confluent to $\rightarrow_{\mathcal{R}}$. A term t is a normal form if there is no term t' such that $t \rightarrow_{\mathcal{R}} t'$. A term t is terminating (strongly normalizing) if there is no infinite reduction sequence starting from term t . A STRS \mathcal{R} is terminating if every term is terminating to $\rightarrow_{\mathcal{R}}$. A STRS \mathcal{R} is complete if \mathcal{R} is confluent and terminating.

If $r \in \mathcal{V}$ then the rewrite rule $l \rightarrow r$ is said to be collapsing. If some variable has more occurrences in r than it has in l then the rewrite rule $l \rightarrow r$ is said to be duplicating. If $l : b$, $r : b'$ and $b \succ b'$ then the rewrite rule $l \rightarrow r$ is said to be decreasing.

When $\mathcal{S} = \{*\}$ with an empty relation, an 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} .

Aoto and Toyama [1], [2] defined the notion of sort attachment and formulated the notion of persistence using sort attachment. We mainly follow basic definitions in [2] in this subsection.

Let \mathcal{F} and \mathcal{V} be sets of function symbols and variables, respectively, on a trivial set $\{*\}$ of sorts with empty relation on it. Terms built from this language are called unsorted terms. Let \mathcal{S} be another set of sorts with well-founded partial ordering \succ on it. \mathcal{F} is a set of arity fixed function symbols.

A sort attachment τ of \mathcal{F} on \mathcal{S} is a family of mapping $\cup_n (\mathcal{F}_n \rightarrow \mathcal{S}^{n+1}) \cup (\mathcal{V} \rightarrow \mathcal{S})$. We can assume that there are countably infinite variables x with $\tau(x) = b$ for each $b \in \mathcal{S}$. $f \in \mathcal{F}$ with $\tau(f) = (b_1, \dots, b_n, b')$ is written as $f: b_1 \times \dots \times b_n \rightarrow b'$. The set of τ -sorted function symbols from \mathcal{F} and that of τ -sorted variables from \mathcal{V} are denoted by \mathcal{F}^τ and \mathcal{V}^τ , respectively. A term t is said to be well-sorted under τ with sort b (written as $\tau(t) = b$) if and only if $t \in \mathcal{T}(\mathcal{F}^\tau, \mathcal{V}^\tau)^b$. The set of well-sorted terms under τ is denoted by \mathcal{T}^τ , i.e. $\mathcal{T}^\tau = \{t \in \mathcal{T} \mid t : b \text{ for some } b \in \mathcal{S}\}$. Clearly, $\mathcal{T}^\tau \subseteq \mathcal{T}$. A term t is said to be strict well-sorted under τ with sort b if and only if $t \in \mathcal{T}(\mathcal{F}^\tau, \mathcal{V}^\tau)^b$ is strict. Well-sorted contexts are defined by special constants \square^{b_1}, \dots . Well-sorted terms and contexts are often treated as sorted terms and contexts, respectively.

Let $(\mathcal{F}, \mathcal{R})$ be a TRS. A sort attachment τ of \mathcal{F} on \mathcal{S} is said to be consistent with \mathcal{R} if and only if for any rewrite rule $l \rightarrow r \in \mathcal{R}$, l and r are strictly well-sorted under τ and $\tau(l) \succeq \tau(r)$. The set of τ -sorted rewrite rules of \mathcal{R} is denoted by \mathcal{R}^τ . Note that \mathcal{R}^τ acts on \mathcal{T}^τ , i.e. well-sorted terms $s, t \in \mathcal{T}^\tau$ whenever $s \rightarrow_{\mathcal{R}^\tau} t$; and that for any $s, t \in \mathcal{T}^\tau$, $s \rightarrow_{\mathcal{R}} t$ if and only if $s \rightarrow_{\mathcal{R}^\tau} t$.

Using the sort attachment, persistence can be alternatively formulated as follows. It is clear that definition of Zantema [23] and the following definition are equivalent when set of sorts with empty relation on it.

Definition 2.1: A property P is persistent if and only if for any TRS $(\mathcal{F}, \mathcal{R})$ and any sort attachment τ that is consistent with \mathcal{R} ,

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

We consider the persistence of termination for TRSs on ordered sorts using definition 2.1 in this paper instead of Zantema's definition. From now on, we assume that a set \mathcal{S} of sorts with well-founded strict partial ordering \succ on it and a TRS \mathcal{R} are given. Then an attachment τ on \mathcal{S} that is consistent with \mathcal{R} is fixed.

III. CHARACTERIZATION OF UNSORTED TERMS

In this section we give a characterization of unsorted terms by ordered sorts. The proofs of the following basic lemmas were given by Aoto and Toyama [2].

Definition 3.1: The top sort (under τ) of an unsorted term t is defined as follows:

- $top(t) = \tau(t)$ if $t \in \mathcal{V}$.
- $top(t) = b'$ if $t = f(t_1, \dots, t_n)$ with $f : b_1 \times \dots \times b_n \rightarrow b'$.

Definition 3.2: Let $t = C[t_1, \dots, t_n]$ ($n \geq 0$) be an unsorted term with $C[\dots] \neq \square$. We write $t = C[t_1, \dots, t_n]$ if and only if

1. $C:b_1 \times \dots \times b_n \rightarrow b'$ is a context that is well-sorted under τ .
2. $top(t_i) \not\prec b_i$ for all $i = 1, \dots, n$.

The t_1, \dots, t_n are said to be the *principal subterms* of t .

We denote $t = C \langle \langle t_1, \dots, t_n \rangle \rangle$ if either $t = C[t_1, \dots, t_n]$ or $C = \square$ and $t_i \in \{t_1, \dots, t_n\}$. Multiset $S(t)$ consists of all principal subterms of $t = C[t_1, \dots, t_n]$.

Definition 3.3: Let t be an unsorted term. *Rank* of t is defined as follows:

- $rank(t) = 1$ if t is well-sorted term.
- $rank(t) = 1 + \max\{rank(t_1), \dots, rank(t_n)\}$ if $t = C[t_1, \dots, t_n]$.

Definition 3.4: Let t be an unsorted term. *Cap* of t is defined as follows:

- $cap(t) = t$ if t is well-sorted term.
- $cap(t) = C[\dots]$ if $t = C[t_1, \dots, t_n]$.

Definition 3.5: A rewrite step $s \rightarrow_{\mathcal{R}} t$ is said to be *inner* (written as $s \rightarrow_{\mathcal{R}}^i t$) if and only if $s = C[s_1, \dots, C'[l\theta], \dots, s_n] \rightarrow_{\mathcal{R}} C[s_1, \dots, C'[r\theta], \dots, s_n] = t$ for some $s_1, \dots, s_n, l \rightarrow r \in \mathcal{R}, \theta$ and C' , otherwise *outer* (written as $s \rightarrow_{\mathcal{R}}^o t$).

Definition 3.6: A rewrite step $s \rightarrow_{\mathcal{R}}^o t$ is said to be *vanishing* if and only if $s = C[s_1, \dots, \theta(x), \dots, s_n] \rightarrow_{\mathcal{R}}^o \theta(x) = t$ for some s_1, \dots, s_n, C and θ such that $C'\theta = C[s_1, \dots, \square, \dots, s_n]$ and $C'[x] \rightarrow x \in \mathcal{R}$.

Definition 3.7: A rewrite rule $l \rightarrow r$ in \mathcal{R} is said to be *decreasing* if and only if $top(l) \succ top(r)$. A rewrite step $s \rightarrow_{\mathcal{R}} t$ is said to be *decreasing* if and only if $top(s) \succ top(t)$.

Definition 3.8: A rewrite step $s \rightarrow_{\mathcal{R}} t$ is said to be *destructive at level 1* if and only if $s \rightarrow^o t$ is either vanishing or decreasing. The rewrite step $s \rightarrow_{\mathcal{R}} t$ is said to be *destructive at level $k + 1$* if and only if $s = C[s_1, \dots, s_j, \dots, s_n] \rightarrow_{\mathcal{R}}^i C[s_1, \dots, t_j, \dots, s_n] = t$ with $s_j \rightarrow_{\mathcal{R}} t_j$ *destructive at level k* .

Lemma 3.9: If a rewrite step $s \rightarrow_{\mathcal{R}}^o t$ is not vanishing then $top(s) \succeq top(t)$. If a rewrite step $s \rightarrow_{\mathcal{R}}^o t$ is not destructive at level 1 then $top(s) = top(t)$.

Lemma 3.10: If $s \rightarrow_{\mathcal{R}} t$ then $rank(s) \geq rank(t)$. If a rewrite step $s \rightarrow_{\mathcal{R}} t$ is vanishing then $rank(s) > rank(t)$.

Definition 3.11: The *grade* ($grade(t) \in N \times S$) of a term t is defined by $grade(t) = \langle rank(t), top(t) \rangle$ where N is the set of all natural numbers.

Let $>$ be the lexicographic ordering on $N \times S$ induced from $>$ on N and \succ on S . The lexicographic ordering $>$ on $N \times S$ is well-founded since orderings $>$ on N and \succ on S are well-founded.

Lemma 3.12: If $s \rightarrow_{\mathcal{R}} t$ then $grade(s) \geq grade(t)$. If a rewrite step $s \rightarrow_{\mathcal{R}} t$ is destructive at level 1 then $grade(s) > grade(t)$.

IV. PERSISTENCE OF TERMINATION

In this section we discuss the persistence of termination for TRSs on ordered sorts and give the example as application

of this result. Furthermore we obtain that completeness is persistent for TRS on ordered sorts. We mainly follow the Ohlebusch's argument in [18].

Lemma 4.1: Let t be a strictly well-sorted term. Suppose $t = C[x_1, \dots, x_n]$ with all variables displayed and $C : b_1 \times \dots \times b_n \rightarrow b'$. Then for any substitution θ , if $x_i = x_j$ and $\theta(x_i)$ is a principal subterm of $t\theta$ then $\theta(x_j)$ is also a principal subterm of $t\theta$.

Let s_1, \dots, s_n and t_1, \dots, t_n be terms. We write $\langle s_1, \dots, s_n \rangle \times \langle t_1, \dots, t_n \rangle$ if and only if for any $1 \leq i, j \leq n$, $s_i = s_j$ implies $t_i = t_j$.

Lemma 4.2: If $C[s_1, \dots, s_n] \rightarrow_{\mathcal{R}}^o C' \langle \langle s_{i_1}, \dots, s_{i_m} \rangle \rangle$ and $\langle s_1, \dots, s_n \rangle \times \langle t_1, \dots, t_n \rangle$ then $C[t_1, \dots, t_n] \rightarrow_{\mathcal{R}}^o C'[t_{i_1}, \dots, t_{i_m}]$.

Lemma 4.3: If $s \rightarrow_{\mathcal{R}} t$ is a non-destructive rewrite step and applied rewrite rule is not duplicating, the multiset inclusion $S(t) \subseteq S(s)$ holds.

Lemma 4.4: If $s = C[s_1, \dots, s_j, \dots, s_n] \rightarrow_{\mathcal{R}}^i C[s_1, \dots, t_j, \dots, s_n] = t$ is destructive at level 2 and $cap(s) \neq cap(t)$ then $S(t) = S(s) \setminus \{s_j\} \cup S(t_j)$. If $s = C[s_1, \dots, s_j, \dots, s_n] \rightarrow_{\mathcal{R}}^i C[s_1, \dots, t_j, \dots, s_n] = t$ is destructive at level 2 and $cap(s) = cap(t)$ then $S(t) = S(s) \setminus \{s_j\} \cup \{t_j\}$.

Lemma 4.5: If $s \rightarrow_{\mathcal{R}}^o t$ is not destructive at level 1 then $cap(s) \rightarrow_{\mathcal{R}} cap(t)$. If $s \rightarrow_{\mathcal{R}}^i t$ is not destructive at level 2 then $cap(s) = cap(t)$.

Proof. Let $s = C[s_1, \dots, s_n] \rightarrow_{\mathcal{R}}^o C' \langle \langle s_{i_1}, \dots, s_{i_m} \rangle \rangle = t$. Since $\langle s_1, \dots, s_n \rangle \times \langle \square, \dots, \square \rangle$ and lemma 4.2, $cap(s) = C[\dots] \rightarrow_{\mathcal{R}} C'[\dots] = cap(t)$. Let $s = C[s_1, \dots, s_j, \dots, s_n] \rightarrow_{\mathcal{R}}^i C[s_1, \dots, t_j, \dots, s_n] = t$ and $s_j \rightarrow_{\mathcal{R}} t_j$. Since $cap(s) = C[\dots]$ and $cap(t) = C[\dots]$, it is trivial. \square

Definition 4.6: For any infinite rewrite sequence $D : s_0 \rightarrow_{\mathcal{R}} s_1 \rightarrow_{\mathcal{R}} s_2 \rightarrow_{\mathcal{R}} \dots$, we define the *grade* of D to be $grade(D) = grade(s_0)$.

Definition 4.7: Let $>$ is the lexicographic ordering on $N \times S$ and $>^{mult}$ is the multiset extension of it. We define $\#s = \{grade(t) \mid t \in S(s)\}$, i.e., $\#s$ denote the multiset of the grades of the principal subterms of s .

Multiset ordering $>^{mult}$ is well-founded since lexicographic ordering $>$ on $N \times S$ is well-founded [5].

Lemma 4.8: If $s \rightarrow_{\mathcal{R}}^i t$ is destructive at level 2 then $\#s >^{mult} \#t$.

Proof. Let s and t be $C[s_1, \dots, s_j, \dots, s_n]$ and $C[s_1, \dots, t_j, \dots, s_n]$, respectively.

1. $cap(s) \neq cap(t)$. By lemma 4.4, $S(t) = S(s) \setminus \{s_j\} \cup S(t_j)$.

- $s_j \rightarrow_{\mathcal{R}} t_j$ is vanishing. Since $rank(s_j) > rank(t_j)$ and $rank(t_j) > rank(u)$ for any $u \in S(t_j)$, $rank(s_j) > rank(u)$ holds. So $grade(s_j) > grade(u)$ for any $u \in S(t_j)$ holds. Hence, $\#s >^{mult} \#t$ holds.
- $s_j \rightarrow_{\mathcal{R}} t_j$ is decreasing. Since $rank(s_j) \geq rank(t_j)$ and $rank(t_j) > rank(u)$ for any $u \in S(t_j)$, $rank(s_j) > rank(u)$ holds. So $grade(s_j) > grade(u)$ for any $u \in S(t_j)$ holds. Hence, $\#s >^{mult} \#t$ holds.

2. $cap(s) = cap(t)$. By lemma 4.4, $S(t) = S(s) \setminus \{s_j\} \cup \{t_j\}$. *Proof.* By lemma 4.10, it is trivial. \square

- $s_j \rightarrow t_j$ is vanishing. Since $rank(s_j) > rank(t_j)$, $grade(s_j) > grade(t_j)$ holds.
- $s_j \rightarrow t_j$ is decreasing. Since $rank(s_j) \geq rank(t_j)$ and $top(s_j) \succ top(t_j)$, $grade(s_j) > grade(t_j)$ holds.

Therefore, $\#s >^{mult} \#t$ holds. \square

Lemma 4.9: Let \mathcal{R}^τ be a terminating STRS. Let $D : s_0 \rightarrow_{\mathcal{R}} s_1 \rightarrow_{\mathcal{R}} s_2 \rightarrow_{\mathcal{R}} \dots$ be an infinite rewrite sequence of minimal grade with respect to \mathcal{R} . Then the following statements hold.

1. There are infinitely many outer rewrite steps in D .
2. There are infinitely many inner rewrite steps in D which are destructive at level 2.
3. There are infinitely many duplicating outer rewrite steps in D .

Proof. Since $grade(s_j) = grade(D)$ for any $j \in \mathbb{N}$, there is no rewrite step which is destructive at level 1.

1. Suppose that there are only finitely many outer rewrite steps in D . Then we can assume that there is no outer rewrite step in D . If $s_0 = C[t_1, \dots, t_n]$ then there must be an infinite rewrite sequence starting from some $t_i \in S(s_0)$. Since $rank(t_i) < rank(s_0)$ this contradicts the minimality of $grade(D)$.

2. Suppose that there are only finitely many inner rewrite steps in D which are destructive at level 2. Then we can assume that there is no inner rewrite step in D which is destructive at level 2. By lemma 4.5, if $s \rightarrow_{\mathcal{R}}^o t$ in D then $cap(s) \rightarrow_{\mathcal{R}^\tau} cap(t)$ and if $s \rightarrow_{\mathcal{R}}^i t$ in D then $cap(s) = cap(t)$. By the case 1, \mathcal{R}^τ is not terminating. This is contradiction by the assumption.

3. Suppose that there are only finitely many outer rewrite steps in which are applied duplicating rules in D . We consider the following cases.

If $s_j \rightarrow_{\mathcal{R}}^o s_{j+1}$ then $S(s_{j+1}) \subseteq S(s_j)$ since the rewrite step is non-destructive and non-duplicating and lemma 4.3. Then $\#s_j \geq^{mult} \#s_{j+1}$ holds.

If $s_j \rightarrow_{\mathcal{R}}^i s_{j+1}$ is not destructive at level 2 then we have $s_j = C[t_1, \dots, t_k, \dots, t_n] \rightarrow_{\mathcal{R}}^i C[t_1, \dots, t'_k, \dots, t_n] = s_{j+1}$ where $t_k \rightarrow_{\mathcal{R}} t'_k$. Then $\#s_j \geq^{mult} \#s_{j+1}$ holds since $grade(t_k) \geq grade(t'_k)$.

If $s_j \rightarrow_{\mathcal{R}}^i s_{j+1}$ is destructive at level 2 then we have $s_j = C[t_1, \dots, t_k, \dots, t_n] \rightarrow_{\mathcal{R}}^i C[t_1, \dots, t'_k, \dots, t_n] = s_{j+1}$ where $t_k \rightarrow_{\mathcal{R}} t'_k$ is destructive at level 1. By lemma 4.8, $\#s_j >^{mult} \#s_{j+1}$. By the well-foundedness of $>^{mult}$, there are only finitely many inner rewrite steps which are destructive at level 2 in D . This contradicts the case 2. \square

Lemma 4.10: Let \mathcal{R}^τ be a terminating STRS. Assume that \mathcal{R} is not terminating. Then \mathcal{R} has duplicating rules and \mathcal{R} has collapsing or decreasing rules.

Proof. By lemma 4.9, it is trivial. \square

Theorem 4.11: The following statements hold.

1. Termination is a persistent property of TRSs on ordered sorts without collapsing and decreasing rules.
2. Termination is a persistent property of TRSs on ordered sorts without duplicating rules.

Example 4.12: We show that the following TRS \mathcal{R} is terminating using theorem 4.11. To show the termination of the following TRS directly seems difficult from known results (E.g. recursive path ordering [7]). Also, we can not use the modularity results for composable TRSs [17], [19] and hierarchical combinations and hierarchical combinations with common subsystem of TRSs [16], [19]. Furthermore, we can not use the Zantema's result [23] for proving termination of the following TRS. However, we can show the termination of next TRS using our results in this paper.

$$\mathcal{R} = \begin{cases} g(x, B) \rightarrow g(x, A) & (r1) \\ g(x, B) \rightarrow d(x, A) & (r2) \\ g(x, d(z, B)) \rightarrow x & (r3) \\ I(A, g(x, d(y, C))) \rightarrow I(B, g(x, d(y, C))) & (r4) \\ I(x, g(x, d(z, z))) \rightarrow d(z, z) & (r5) \\ d(z, A) \rightarrow e(z, C) & (r6) \end{cases}$$

Let $S = \{0, 1, 2\}$, $1 \succ 0$ and $2 \succ 0$.

$$\tau = \begin{cases} g : 0 \times 0 \rightarrow 1 \\ I : 0 \times 1 \rightarrow 2 \\ d : 0 \times 0 \rightarrow 0 \\ e : 0 \times 0 \rightarrow 0 \\ A : 0, B : 0, C : 0 \end{cases}$$

Any well-sorted term in \mathcal{T}^0 , \mathcal{T}^1 and \mathcal{T}^2 is terminating, i.e. any well-sorted term in \mathcal{T}^τ is terminating. We consider the following cases:

- $t \in \mathcal{T}^0$. Then (r6) is the only applicable rule. A TRS $\{(r6)\}$ is terminating using recursive path ordering. Hence, t is terminating.
- $t \in \mathcal{T}^1$. Then (r1), (r2), (r3) and (r6) are the only applicable rules. A TRS $\{(r1), (r2), (r3), (r6)\}$ is terminating using recursive path ordering. Hence, t is terminating.
- $t \in \mathcal{T}^2$. Then (r1), (r2), (r3), (r4), (r5) and (r6) are the applicable rules. For any proper subterm s of t , $top(s) = 0$ or $top(s) = 1$. Since the above two cases, s is terminating. Since $top(t) = 2$, (r4) and (r5) are the only applicable rules to root position of term t . Hence, t is terminating.

Then STRS \mathcal{R}^τ is terminating. Since \mathcal{R}^τ has no duplicating rules and theorem 4.11, \mathcal{R} is terminating.

Furthermore we obtain the persistence of completeness for TRSs on ordered sorts. The following theorem was given by Aoto and Toyama [2].

Theorem 4.13: ([2]) Confluence is a persistent property of TRSs on ordered sorts.

Since a complete TRS is confluent and terminating, we obtain the following corollary from theorem 4.11 and theorem 4.13.

Corollary 4.14: The following statements hold.

1. Completeness is a persistent property of TRSs on ordered sorts without collapsing and decreasing rules.
2. Completeness is a persistent property of TRSs on ordered sorts without duplicating rules.

In this paper, we have shown that the Zantema's result [23] is preserved for Aoto and Toyama's extension [2] in the subclass of order sorted term rewriting systems. That is, we have shown that termination is persistent for TRSs on ordered sorts without collapsing, decreasing or duplicating rules. Furthermore, we have given the example as application of our results. Also we obtain the persistence of completeness for TRSs on ordered sorts.

However, the difference between our work and Ohsaki's work [21] is still unclear. So we consider this point for the future.

- [22] G. Smolka, W. Nutt, J. A. Goguen and J. Meseguer, "Order-sorted equational computation, resolution of equations in algebraic structures," vol.2, pp.297–367, Academic Press, 1989.
- [23] H. Zantema, "Termination of term rewriting: interpretation and type elimination," *J. Symbolic Computation*, 17, pp.23–50, 1994.

REFERENCES

- [1] T. Aoto and Y. Toyama, "Persistency of confluence," *J. Universal Computer Science*, 3, pp.1134–1147, 1997.
- [2] T. Aoto and Y. Toyama, "Extending persistency of confluence with ordered sorts," *Research report*, IS-RR-96-0025F, School of Information Science, JAIST, 1996.
- [3] T. Aoto, "Solution to the problem of Zantema on a persistent property of term rewriting systems," *Proc. 7th International Conf. on Algebraic and Logic Programming*, LNCS, 1490, Springer-Verlag, pp.250–265, 1998.
- [4] F. Baader and T. Nipkow, "Term rewriting and all that," Cambridge University Press, 1998.
- [5] N. Dershowitz and Z. Manna, "Proving termination with multiset orderings," *Commun. ACM*, 22 (8), pp.465–476, 1979.
- [6] N. Dershowitz and J. P. Jouannaud, "Rewrite systems," In *Handbook of Theoretical Computer Science*, vol.B, ed. J. van Leeuwen, pp.243–320, The MIT Press/Elsevier, 1990.
- [7] N. Dershowitz, "Termination of rewriting," *J. Symbolic Computation*, 3, pp.69–116, 1987.
- [8] J. A. Goguen and J. Meseguer, "Order-sorted algebra 1: equational deduction for multiple inheritance, overloading, exceptions and partial operations" *Theoretical Computer Science*, 105, (2), pp.217–273, 1992.
- [9] M. Iwami and Y. Toyama, "On the persistency of termination of term rewriting systems with ordered sorts," *Proc. 14th Conf. on Japan Society for Software Science and Technology*, pp.357–360, 1997 (in Japanese).
- [10] M. Iwami, "Termination of higher-order rewrite systems," Ph.D. thesis, JAIST, 1999.
- [11] M. Iwami, "Persistence of termination for non-overlapping term rewriting systems," *Proc. International Conf. on Information Technology*, to appear.
- [12] M. Iwami, "Persistence of termination for locally confluent overlay term rewriting systems," *Proc. International Conf. on Information Technology*, to appear.
- [13] M. Iwami, "Persistence of termination for right-linear overlay term rewriting systems," *Proc. International Conf. on Information Technology*, to appear.
- [14] M. Iwami, "Persistence of semi-completeness for term rewriting systems," *Proc. International Conf. on Information Technology*, to appear.
- [15] J. W. Klop, "Term rewriting systems," In *Handbook of Logic in Computer Science*, vol.2, pp.1–112, ed. S. Abramsky, D. Gabbay and T. Mabiliaum, Oxford University Press, 1992.
- [16] M.R.K. Krishna Rao, "Modular proofs for completeness of hierarchical term rewriting systems," *Theoretical Computer Science*, 151, pp.487–512, 1995.
- [17] E. Ohlebusch, "Modular properties of composable term rewriting systems," *J. Symbolic Computation*, 20, (1), pp.1–41, 1995.
- [18] E. Ohlebusch, "A simple proof of sufficient conditions for the termination of the disjoint union of term rewriting systems," *Bullen of the EATCS*, 49, pp.178–183, 1993.
- [19] E. Ohlebusch, "Advanced topics in term rewriting," Springer-Verlag, 2002.
- [20] H. Ohsaki and A. Middeldorp, "Type introduction for equational rewriting," *Proc. 4th International Symp. on Logical Foundations of Computer Science*, LNCS, 1234, Springer-Verlag, pp.283–293, 1997.
- [21] H. Ohsaki, "Termination of term rewriting systems: transformation and persistence," Ph.D. thesis, Tsukuba University, 1998.