

# The Longest Perpetual Reductions in Orthogonal Expression Reduction Systems

Zurab Khasidashvili  
School of Information Systems, UEA  
Norwich NR4 7TJ England  
zurab@sys.uea.ac.uk\*

## Abstract

We consider reductions in Orthogonal Expression Reduction Systems (OERS), that is, Orthogonal Term Rewriting Systems with bound variables and substitutions, as in the  $\lambda$ -calculus. We design a strategy that for any given term  $t$  constructs a longest reduction starting from  $t$  if  $t$  is strongly normalizable, and constructs an infinite reduction otherwise. The Conservation Theorem for OERSs follows easily from the properties of the strategy. We develop a method for computing the length of a longest reduction starting from a strongly normalizable term. We study properties of pure substitutions and several kinds of similarity of redexes. We apply these results to construct an algorithm for computing lengths of longest reductions in *strongly persistent* OERSs that does not require actual transformation of the input term. As a corollary, we have an algorithm for computing lengths of longest developments in OERSs.

## 1 Introduction

A strategy is *perpetual* if, given a term  $t$ , it constructs an infinite reduction starting from  $t$  whenever such a reduction exists, that is, whenever  $t$  is not strongly normalizable. Perpetual strategies are mostly interesting because termination of a perpetual reduction (constructed according to a perpetual strategy) implies strong normalization of the initial term. For orthogonal (left-linear and non-overlapping) TRSs a very simple perpetual strategy exists — just contract any innermost redex (O’Donnell [15]). In fact, any *complete* strategy, i.e., a strategy that in each term contracts a redex that does not erase any other redex, is perpetual. Moreover, one can even reduce redexes all erased arguments of which are strongly normalizable (Klop [11]).

It is easy to see that in any infinite reduction a redex that itself has an infinite reduction, call it an *infinite* redex, is contracted. Thus in order to construct an infinite reduction one should try to retain at least one *potentially infinite* redex — a subterm that can become an infinite redex (more precisely, that has a descendant under some reduction that is an infinite redex). Thus any strategy that does not erase potentially infinite redexes is perpetual. In OTRSs, any potentially infinite redex necessarily has an infinite reduction. That is why all the above strategies are perpetual. In orthogonal Expression Reduction Systems (that is, TRSs with bound variables and substitution

---

\*In: Proc. of the 3<sup>rd</sup> International Conference on Logical Foundations of Computer Science, ‘Logic at St. Petersburg’, LFCS’94, Springer LNCS, vol. 813, Narode A., Matiyasevich Yu. V. eds. St. Petersburg, 1994. p. 191-203.

mechanism [6]) a strongly normalizable subterm may also be a potentially infinite redex — after contraction of an outer redex a term can be substituted in it that makes the subterm no longer strongly normalizable. Thus innermost reductions and complete reductions are no longer perpetual in OERSs. Therefore one can erase only strongly normalizable arguments in which no substitution of external subterms is possible. For the lambda-calculus, such a strategy was found by Barendregt et al. [2].

In this paper, we design a perpetual strategy for all OERSs. It is a generalization of the perpetual strategy of the  $\lambda$ -calculus and of the *limit* strategy [8]. Our aim is not only to construct an infinite reduction of any given term  $t$  whenever it exists, but also to construct a longest possible one if  $t$  is strongly normalizable and to find a method to characterize the lengths of longest reductions. The idea is that, as mentioned above, in order to construct perpetual reductions one should try to avoid erasure of subterms in which substitution of terms is possible during reductions of outer redexes. On the other hand, in order to construct a longest possible reductions one should delay contraction of a redex until it will no longer be possible to copy it by reducing an outer redex. The two conditions agree if in each term  $s$  one contracts a *limit* redex, which is defined as follows: choose in  $s$  an *unabsorbed* redex  $u_1$ , i.e., a redex whose descendants never appear inside redex-arguments; choose an erased argument  $s_1$  of  $u_1$  that is not in normal form; choose in  $s_1$  an unabsorbed redex  $u_2$ , and so on, as long as possible. The last chosen redex is a limit redex of  $s$ .

In non-left-absorbing systems, where no subterms to the left of a contracted redex can appear (i.e., be absorbed) in the arguments of created redexes, the leftmost-outermost redexes are unabsorbed. In particular, the  $\lambda$ -calculus, Combinatory Logic, and all left-normal systems are non-left-absorbing. Therefore the perpetual strategy of Barendregt et al. [2] is a limit strategy. The Conservation Theorem for the  $\lambda$ -calculus [2, 1] states that non-erasing redexes are perpetual. Its proof remains valid for OERSs if one uses the limit strategy instead of the perpetual strategy of [2].

Our method for proving that the reductions constructed according to our perpetual strategy are the longest, and for computing their lengths, is a refinement of Nederpelt-Klop method [14, 10], used to reduce proofs of strong normalization to proofs of weak normalization. For any OERS  $R$ , we define the corresponding non-erasing OERS  $R_\mu$ , called the  $\mu$ -*extension* of  $R$ . We add fresh function symbols  $\mu^n$  in the alphabet of  $R$ . For any  $R$ -rule  $r : t \rightarrow s$ , we keep the erased arguments of  $t$  in the right-hand side of each corresponding  $R_\mu$ -rule  $r_\mu : t' \rightarrow s'$  as  $\mu$ -*erased* arguments of  $s'$ . Since this transformation affects the structure of redex-creation in  $R$ , and since erasure of arguments of a redex depends not only on the rule, but the arguments themselves, we have to introduce infinitely many  $R_\mu$ -rules for each  $R$ -rule. This helps to give a natural translation of  $R$ -reductions into  $R_\mu$ -reductions and vice-versa. Finally, we also keep all  $\mu$ -symbols of  $t'$  as  $\mu$ -erased symbols in  $s'$ , since they can be used as *counters* of steps in longest normalizing reductions. We then show that a term  $o$  in an OERS  $R$  is strongly normalizable if it is weakly normalizable in  $R_\mu$ , and that the least upper bound of lengths of  $R$ -reductions starting from  $o$  coincides with the number of  $\mu$ -occurrences in the  $R_\mu$ -normal form of  $o$ .

Unabsorbed redexes exist in any term not in a normal form, but they can be found efficiently only in strongly sequential OERSs [5]. Therefore, our method is applicable to all strongly sequential OERSs that are strongly normalizing. Several typed  $\lambda$ -calculi and typed OTRSs are such OERSs. Moreover, it is not always necessary to do actual transformation of the input term in order to compute the lengths of longest reductions. We show this for the case of *strongly persistent* OERSs. In such OERSs, creation of redexes is not possible during ‘pure substitution steps’; creation is only possible during the ‘TRS part’ of reduction steps, and the arguments of a contracted redex and the

context in which the reduction takes place do not take part in the creation. This kind of creation we call *generation*. We define several notions to characterize similarity of redexes in OERSs. The above result relies on the fact that *strongly similar* redexes generate the same number of strongly similar redexes.

The results of this paper with complete proofs are reported in [9].

## 2 Orthogonal Expression Reduction Systems

Klop introduced *Combinatory Reduction Systems (CRSs)* in [10] to provide a uniform framework for reductions with substitutions (also referred to as higher order rewriting) as in the  $\lambda$ -calculus [1]. Several other formalisms have been introduced later. We refer to Klop et al. [12] for a survey. Here we use a system of higher order rewriting, *Expression Reduction Systems (ERSs)*, defined in [6] (under the name of CRSs).

**Definition 2.1** (1) Let  $\Sigma$  be an *alphabet*, comprising *variables*  $v_0, v_1, \dots$ ; *function symbols*, also called *simple operators*; and *operator signs* or *quantifier signs*. Each function symbol has an *arity*  $k \in \mathbb{N}$ , and each operator sign  $\sigma$  has an *arity*  $(m, n)$  with  $m, n \neq 0$  such that, for any sequence  $x_1, \dots, x_m$  of pairwise distinct variables,  $\sigma x_1 \dots x_m$  is a *compound operator* or a *quantifier* with *arity*  $n$ . Occurrences of  $x_1, \dots, x_m$  in  $\sigma x_1 \dots x_m$  are called *binding variables*. Each quantifier  $\sigma x_1 \dots x_m$  has a *scope indicator*  $(k_1, \dots, k_l)$  to specify the arguments in which  $\sigma x_1 \dots x_m$  binds all free occurrences of  $x_1, \dots, x_m$ . *Terms* are constructed from variables using functions and quantifiers in the usual way.

(2) *Metaterms* are constructed from *terms*, *term metavariables*  $A, B, \dots$ , which range over terms, and *object metavariables*  $a, b, \dots$ , which range over variables. Besides the usual rules for term-formation, one is allowed to have *metasubstitutions* — expressions of the form  $(A_1/a_1, \dots, A_n/a_n)A_0$ , where  $a_i$  are object metavariables and  $A_j$  are metaterms. Metaterms that do not contain metasubstitutions are called *simple metaterms*. An *assignment* maps each object metavariable to a variable and each term metavariable to a term over  $\Sigma$ . If  $t$  is a metaterm and  $\theta$  is an assignment, then the  $\theta$ -*instance*  $t\theta$  of  $t$  is the term obtained from  $t$  by replacing metavariables with their values under  $\theta$ , and by replacing subterms of the form  $(t_1/x_1, \dots, t_n/x_n)t_0$  by the result of substitution of terms  $t_1, \dots, t_n$  for free occurrences of  $x_1, \dots, x_n$  in  $t_0$ .

**Definition 2.2** (I) An *Expression Reduction System (ERS)* is a pair  $(\Sigma, R)$ , where  $\Sigma$  is an *alphabet*, described in Definition 2.1, and  $R$  is a set of *rewrite rules*  $r : t \rightarrow s$ , where  $t$  and  $s$  are metaterms such that  $t$  is a simple metaterm and is not a metavariable, and each term metavariable that occurs in  $s$  occurs also in  $t$ . Further:

(1) The metaterms  $t$  and  $s$  do not contain variables, and each occurrence of an object metavariable in  $t$  and  $s$  is bound. The metaterm  $s$  may contain occurrences of object metavariables that do not occur in  $t$ . They are called *additional object metavariables*.

(2) Each rule  $r : t \rightarrow s$  has a set of *admissible assignments*  $AA(r)$  such that, for any assignment  $\theta \in AA(r)$ :

(a) occurrences of variables in  $s\theta$  that correspond to additional object metavariables of  $s$  do not bind variables in subterms that correspond to term metavariables of  $s$ .

(b) For any term metavariable  $A$  and any object metavariable  $a$  occurring in  $t$  or  $s$ , an occurrence of  $A\theta$  in  $s\theta$  is in the scope of an occurrence of  $a\theta$  in  $s\theta$  iff any occurrence of  $A\theta$  in  $t\theta$  is in the scope of an occurrence of  $a\theta$  in  $t\theta$ .

(c) For any rule  $r : t \rightarrow s$  in  $R$  and any assignment  $\theta \in AA(r)$ ,  $t\theta$  is an  $r$ -redex or an  $R$ -redex, and  $s\theta$  is the *contractum* of  $t\theta$ . Redexes that are instances of the left-hand side of the same rule (i.e., with the same set of admissible substitutions) are called *weakly similar*.

(II)  $R$  is *simple* if right-hand sides of  $R$ -rules are simple metaterms.

Our syntax is close to the syntax of the  $\lambda$ -calculus and of First Order Logic. For example, the  $\beta$ -rule is written as  $\beta : Ap(\lambda aA, B) \rightarrow (B/a)A$ , where  $a$  is to be instantiated by a variable and  $A$  and  $B$  are to be instantiated by terms. The expression  $(B/a)A$  is a *metasubstitution*, and its instance  $(t/x)s$  denotes the result of substitution of the term  $t$  for  $x$  in the term  $s$ . To express ‘pure’ substitutions syntactically (instead of metasyntactically) we use  $S$ -reduction rules  $S^{n+1}a_1 \dots a_n A_1 \dots A_n A_0 \rightarrow (A_1/a_1, \dots, A_n/a_n)A_0$ ,  $n = 1, 2, \dots$ , where  $S^{n+1}$  is the *operator sign of substitution* with arity  $(n, n + 1)$  and scope indicator  $(n + 1)$ , and  $a_1, \dots, a_n$  and  $A_1, \dots, A_n, A_0$  are pairwise distinct object and term metavariables, respectively. Thus  $S^{n+1}$  binds free variables only in the last argument. The difference with  $\beta$ -rules is that  $S$ -reductions can only perform  $\beta$ -developments of  $\lambda$ -terms [6].

Below we ignore questions relating to renaming of bound variables.

**Notation** We use  $a, b$  for object metavariables,  $A, B$  for term metavariables,  $c, d$  for constants,  $t, s, e, o$  for terms and metaterms,  $u, v, w$  for redexes, and  $P, Q$  for reductions. We write  $s \subseteq t$  if  $s$  is a subterm of  $t$ . A one-step reduction in which a redex  $u$  in a term  $t$  is contracted is written as  $t \xrightarrow{u} s$  or  $t \rightarrow s$ . We write  $P : t \rightarrow s$  if  $P$  denotes a reduction of  $t$  to  $s$ .  $|P|$  denotes the length of  $P$ .  $P + Q$  denotes the concatenation of  $P$  and  $Q$ .

For any ERS  $R$ ,  $R_f$  is the ERS obtained from  $R$  by adding symbols  $\underline{S}_{n+1}$  in the alphabet and by replacing in right-hand sides of the rules all metasubstitutions of the form  $(t_1/a_1, \dots, t_n/a_n)t$  by  $\underline{S}^{n+1}a_1 \dots a_n t_1 \dots t_n t$ , respectively; If  $R$  is simple, then  $R_{fS} =_{def} R_f =_{def} R$ . Otherwise  $R_{fS} =_{def} R_f \cup \underline{S}$ , where  $\underline{S}$ -rules are obtained from  $S$ -rules by underlining the  $S$ -symbols. For each step  $e = C[t_i\theta] \xrightarrow{u} C[s_i\theta] = o$  in  $R$  there is a reduction  $P : e = C[t_i\theta] \rightarrow C[s'_i\theta] \rightarrow C[s\theta] = o$  in  $R_{fS}$ , where  $C[s'_i\theta] \rightarrow C[s\theta]$  is the rightmost innermost normalizing  $\underline{S}$ -reduction. We call  $P$  the *refinement* of  $u$ . The notion of *refinement* generalizes to  $R$ -reductions with 0 or more steps.

Let  $t \xrightarrow{u} s$  be an  $R_f$ -reduction step and let  $e$  be the contractum of  $u$  in  $s$ . For each argument  $o$  of  $u$  there are 0 or more arguments of  $e$ . We call them  $(u)$ -*descendants* of  $o$ . We refer to the  $i$ -th (from the left,  $i > 0$ ) descendant of  $o$  also as  $(u, i)$ -*descendant* of  $o$ . Correspondingly, subterms of  $o$  have 0 or more *descendants*. By definition, the *descendant*, referred to also as  $(u, @)$ -*descendant*, of each pattern-subterm (i.e., a subterm headed at the pattern) of  $u$  is  $e$ . It is clear what is to be meant by *descendants* of a subterm  $s' \subseteq t$  that is not in  $u$ . We call them also  $(u, *)$ -*descendants* of  $s'$ . In an  $S$ -reduction step  $C[S^{n+1}x_1 \dots x_n t_1 \dots t_n t_0] \xrightarrow{u} C[(t_1/x_1, \dots, t_n/x_n)t_0]$ , the argument  $t_i$  and subterms in  $t_i$  have the same number of descendants as the number of free occurrences of  $x_i$  in  $t_0$ ; the  $i$ -th descendant is referred to as the  $(u, i)$ -*descendant*. All subterms of  $t_0$ , including free occurrences of  $x_1, \dots, x_n$ , have exactly one descendant, the  $(u, 0)$ -*descendant*. The descendant or  $(u, @)$ -*descendant* of the contracted redex  $u$  itself is its contractum. The pairs  $(u, i)$ ,  $(u, @)$ , and  $(u, *)$  are called the *indexes* of corresponding descendants. The descendants of all redexes except the contracted one are called *residuals*.

The notions of *descendant* and *residual* extend by transitivity to arbitrary  $R_{fS}$ -reductions; *indexes* of descendants and residuals are sequences of indexes of (immediate) descendants and residuals in the chain leading from initial to the final subterm. If  $P$  is an  $R$ -reduction, then  $P$ -descendants and  $P$ -residuals are defined to be the descendants and residuals under the refinement

of  $P$ . The *ancestor* relation is converse to the descendant relation. We call the co-initial reductions  $P : t \rightarrow s$  and  $Q : t \rightarrow e$  *strictly equivalent* (written  $P \approx_{st} Q$ ) if  $s = e$  and  $P$ -descendants and  $Q$ -descendants of any subterm of  $t$  are the same in  $s$  and  $e$ .

An ERS is *orthogonal* (OERS) if in no term redex-patterns can overlap and, for any reduction step  $t \xrightarrow{w} s$ , if  $u \subseteq s$  is a descendant of a redex  $v \neq w$  of  $t$ , then  $u$  is weakly similar to  $v$ . The descendant relation and the following *Strict Church-Rosser* theorem is vital in the proofs throughout this paper: for any co-initial reductions  $P$  and  $Q$  in an OERS  $R$  there are reductions  $P'$  and  $Q'$  such that  $P + P' \approx_{st} Q + Q'$  [6].

### 3 Properties of $S$ -reductions

In this section, we study some properties of substitutions.

We call a subterm  $s$  in  $t$  *essential* (written  $ES(s, t)$ ) if  $s$  has at least one descendant under any reduction starting from  $t$  and call it *inessential* (written  $IE(s, t)$ ) otherwise [7]. The notion of essentiality is a generalization of the notion of *neededness* [5, 13] in a way that it works for all subterms, bound variables in particular.

**Notation** Below  $FV(t)$  (resp.  $EFV_R(t)$ ) denotes the set of variables having ( $R$ -essential) free occurrences in  $t$ . For any  $s \subseteq t$ ,  $BV_R(s)$  (resp.  $EBV_R(s)$ ) denotes the set of ( $R$ -essential) free occurrences of  $s$  bound by quantifiers belonging to patterns of  $R$ -redexes that are outside  $s$ .

We call a subterm  $e$  in  $u = Sx_1 \dots x_n t_1 \dots t_n t_0$   *$u$ -inessential* (written  $IE(u; e)$ ) if  $e$  is in  $t_i$  for some ( $1 \leq i \leq n$ ) and  $x_i$  does not have a free occurrence in the  $S$ -normal form of  $t_0$ . The following lemma is easy to prove using the fact that if  $P : t \rightarrow t'$  and  $s \subseteq t$ , then  $IE(s, t)$  iff any  $P$ -descendant of  $s$  is inessential in  $t'$ .

- Lemma 3.1** (1) Let  $u = Sx_1 \dots x_n t_1 \dots t_n t_0 \subseteq t$ . Then  $IE_S(u; t_i)$  iff  $x_i \notin EFV_S(t_0)$ .  
(2) Let  $s \subseteq t$ . Then  $IE_S(s, t)$  iff  $IE_S(u; s)$  for some  $S$ -redex  $u$  in  $t$ .  
(3) Let  $e \subseteq s \subseteq t$ . Then  $ES_S(e, t)$  iff  $ES_S(e, s)$  and  $ES_S(s, t)$ .

**Lemma 3.2** Let  $s$  be obtained from  $t$  by replacing non-overlapping proper subterms  $t_1, \dots, t_n$  with  $s_1, \dots, s_n$ , respectively, where  $s_i$  and  $t_i$  do not contain  $S$ -redexes, and let  $ES_S(s_i, s) \Rightarrow BV_S(t_i) \subseteq BV_S(s_i)$  ( $i = 1, \dots, n$ ). Further, let  $s'$  and  $t'$  be any corresponding subterms in  $s$  and  $t$  that are not in replaced subterms. Then  $IE_S(s', s) \Rightarrow IE_S(t', t)$ .

**Proof** By induction on the length of  $s$ . If  $t$  and  $s$  are not  $S$ -redexes, then the lemma follows easily from Lemma 3.1.(2) and the induction assumption. So suppose that  $t = Sx_1 \dots x_m e_1 \dots e_m e_0$ ,  $s = Sx_1 \dots x_m o_1 \dots o_m o_0$ ,  $s' \subseteq o_l$ , and  $IE_S(s', s)$ . If  $IE_S(s', o_l)$ , then by the induction assumption  $IE_S(t', e_l)$  and hence  $IE_S(t', t)$ . Otherwise, by Lemma 3.1.(3), we have  $IE_S(o_l, s)$ . Hence, by Lemma 3.1.(2),  $IE_S(s; o_l)$ . Thus, by Lemma 3.1.(1),  $x_l \notin EFV_S(o_0)$ . Let us show that  $x_l \notin EFV_S(e_0)$ . By Lemma 3.1.(2), if  $s_i \subseteq o_0$ , then  $ES_S(s_i, s)$  iff  $ES_S(s_i, o_0)$ . Hence, for any  $S$ -essential subterm  $s_i$ ,  $BV_S(t_i) \subseteq BV_S(s_i)$ . By the induction assumption, if  $x_l$  has an  $S$ -essential occurrence in  $e_0$  outside of replaced subterms, then the corresponding occurrence of  $x_l$  in  $o_0$  is  $S$ -essential. If  $x_l$  has an  $S$ -essential occurrence in a subterm  $t_j \subseteq e_0$ , then, by Lemma 3.1.(3),  $ES_S(t_j, e_0)$ . By the induction assumption,  $ES_S(s_j, o_0)$ . Hence  $BV_S(t_j) \subseteq BV_S(s_j)$ . Thus  $x_l$  has a free occurrence in  $s_j$ . Since  $s_j$  does not contain  $S$ -symbols, it follows from Lemma 3.1.(2) that this occurrence is  $S$ -essential in  $s_j$  and hence, by Lemma 3.1.(3) and  $ES_S(s_j, o_0)$ , is  $S$ -essential in  $o_0$ . Hence  $x_l \notin EFV_S(e_0)$  and, by Lemma 3.1.(1),  $IE_S(t; e_l)$ . Therefore, by Lemma 3.1.(2),  $IE_S(t; t')$  and  $IE_S(t', t)$ .

**Lemma 3.3** Let  $u = C[t_1, \dots, t_n]$  and  $v = C[s_1, \dots, s_n]$  be weakly similar redexes, let  $s_{m_1}, \dots, s_{m_l}$  be the  $v$ -main arguments,  $s_{n_1}, \dots, s_{n_k}$  be the  $v$ -erased arguments, and let for each  $j = 1, \dots, l$  :  $BV(t_{m_j}) \subseteq BV(s_{m_j})$ . Then  $t_{n_1}, \dots, t_{n_k}$  are (not necessarily all)  $u$ -erased arguments.

**Proof** Let  $u \rightarrow t \rightarrow o$  and  $v \rightarrow s \rightarrow e$  be refinements of  $u$  and  $v$ , respectively. Then  $s$  can be obtained from  $t$  by replacing descendants of  $t_1, \dots, t_n$  with  $s_1, \dots, s_n$ . Further, all descendants of  $s_{n_1}, \dots, s_{n_k}$  are  $\underline{S}$ -inessential. Therefore if  $t'_i$  and  $s'_i$  are corresponding descendants of  $t_i$  and  $s_i$  in  $t$  and  $s$ , then  $ES_{\underline{S}}(s'_i, s) \Rightarrow BV_{\underline{S}}(t'_i) \subseteq BV_{\underline{S}}(s'_i)$ . Thus the lemma follows from Lemma 3.2.

## 4 A Perpetual Strategy for OERSs

In this section, we design a strategy that for a term  $t$  in an OERS constructs a longest reduction when  $t$  is strongly normalizable, and constructs an infinite reduction otherwise. We use the strategy to generalize the Conservation Theorem [2, 1] to OERSs. We give also a method to determine the lengths of longest reductions of strongly normalizable terms.

**Definition 4.1** The  $\mu$ -extension  $(\Sigma_\mu, R_\mu)$  of an OERS  $(\Sigma, R)$  is defined as follows:

1.  $\Sigma_\mu = \Sigma \cup \{\mu^n \mid n = 0, 1, \dots\}$ , where  $\mu^n$  is a fresh  $n$ -ary function symbol. For any subterm  $s = \mu^{n+1}(t_1, \dots, t_n, t_0)$  of a term  $t$  over  $\Sigma_\mu$ , the arguments  $t_1, \dots, t_n$ , as well as subterms and symbols in  $t_1, \dots, t_n$  and the head-symbol  $\mu$  itself, are called  $\mu$ -erased or more precisely  $\mu'$ -erased, where  $\mu'$  is the occurrence of the head symbol of  $s$  in  $t$ . The argument  $t_0$  is called  $\mu'$ -main. Symbols and subterms in  $t$  that are not  $\mu$ -erased are called  $\mu$ -main. We denote by  $[t]_\mu$  the term obtained from  $t$  by removing all  $\mu$ -erased symbols.

2.  $R_\mu$  is the set of all rules of the form  $r_\mu : t' \rightarrow s'$  such that

(a) there is a rule  $r : t \rightarrow s$  in  $R$  such that  $[t']_\mu = t$ ;

(b)  $t'$  is linear; its head symbol is not a  $\mu$ -symbol; the  $\mu$ -erased arguments of each  $\mu$ -occurrence  $\mu'$  in  $t'$  are term metavariables, and the  $\mu'$ -main argument is not a term metavariable;

(c) Let  $A_1, \dots, A_n$  be all  $\mu$ -main term metavariables of  $t'$ ,  $B_1, \dots, B_j$  be all  $\mu$ -erased term metavariables of  $t'$ , and let  $k$  be the number of occurrences of  $\mu$ -symbols in  $t'$ . Then

$$s' = \mu^m(\overbrace{\mu^0, \dots, \mu^0}^k, B_1, \dots, B_j, A_{i_1}, \dots, A_{i_l}, s).$$

3. An assignment  $\theta$  is admissible for  $r_\mu$  iff the arguments  $A_{i_1}\theta, \dots, A_{i_l}\theta$  of the redex  $t'\theta$  do not have descendants under the reduction step  $t\theta \rightarrow s\theta$ ; and the assignment  $\theta_\mu$  such that  $A\theta_\mu = [A\theta]_\mu$  and  $a\theta_\mu = a\theta$  for any term metavariable  $A$  and object metavariable  $a$  is admissible for  $r$ .

4. For any  $r_\mu$ -redex  $u = t'\theta$ , we call arguments that correspond to  $A_{i_1}, \dots, A_{i_l}$  *quasi-erased* arguments of  $u$ , and call the arguments that correspond to other metavariables from  $A_1, \dots, A_n$  *quasi-main*.

For example,  $Ap(\mu^3(A, B, \mu^2(C, \lambda a D)), \mu^1(E)) \rightarrow \mu^7(\mu^0, \mu^0, \mu^0, A, B, C, (E/a)D)$  and  $Ap(\mu^3(A, B, \mu^2(C, \lambda a D)), \mu^1(E)) \rightarrow \mu^8(\mu^0, \mu^0, \mu^0, A, B, C, E, (E/a)D)$  are two  $\beta_\mu$ -rules with the same left-hand and different right-hand sides. The arguments  $A, B$ , and  $C$  are  $\mu$ -erased, and  $E$  and  $D$  are  $\mu$ -main. An assignment  $\theta$  is admissible for the first rule iff  $a\theta \in FV(D\theta)$  and is admissible for the second one otherwise.

**Notation**  $\|t\|_\mu$  denotes the number of occurrences of  $\mu$ -symbols in  $t$ .

One can prove using Definition 4.1 that  $R_\mu$  is orthogonal. Let  $t$  be a term over  $\Sigma_\mu$  whose head-symbol is not a  $\mu$ -symbol. Then  $t$  is an  $r_\mu$ -redex (in  $R_\mu$ ) iff  $[t]_\mu = s$  is an  $r$ -redex (in  $R$ ). Moreover, if  $t'$  is the contractum of  $t$  in  $R_\mu$  and  $s'$  is the contractum of  $s$  in  $R$ , then  $[t']_\mu = s'$ . Hence, for any  $R$ -reduction  $Q : s_0 \xrightarrow{u_0} s_1 \xrightarrow{u_1} \dots$  and for any term  $t_0$  in  $R_\mu$  such that  $[t_0]_\mu = s_0$ , there is a reduction  $Q_\mu : t_0 \xrightarrow{v_0} t_1 \xrightarrow{v_1} \dots$  in  $R_\mu$  such that  $[t_i]_\mu = s_i$ , and  $u_i$  and  $v_i$  are corresponding subterms in  $s_i$  and  $t_i$  ( $i = 0, 1, \dots$ ). If  $t_0$  has the  $R_\mu$ -normal form  $t^*$ , then  $t_i \rightarrow t^*$  in  $R_\mu$  and  $i \leq \|t_i\|_\mu \leq \|t^*\|_\mu$ ; hence  $s_0$  is normalizable in  $R$  and  $R_\mu$ . Thus

**Lemma 4.1** (Klop [10]) Let  $t$  be a term in an OERS  $R$ . If  $t$  is weakly normalizable in  $R_\mu$ , then  $t$  is strongly normalizable in  $R_\mu$  and  $R$ .

**Proposition 4.1** (1) Let  $R$  be an OERS,  $u$  and  $v$  be  $R_\mu$ -redexes such that  $u$  is in an argument of  $v$ , and let  $v \xrightarrow{u} w$  in  $R_\mu$ . Then  $w$  is an  $R_\mu$ -redex weakly similar to  $v$ , and quasi-main sequences of  $v$  and  $w$  coincide.

(2) Let  $u$  be a redex in an OERS  $R$  and  $v$  be an  $R_\mu$ -redex such that  $[v]_\mu = u$  and the sets of free variables of quasi-main arguments of  $v$  coincide with that of corresponding arguments of  $u$ . Then an argument of  $v$  is quasi-erased iff the corresponding argument of  $u$  is erased.

(3) Let  $u$  be a redex in an OERS  $R$  and  $v$  be an  $R_\mu$ -redex such that  $[v]_\mu = u$ . Then the corresponding argument of any quasi-erased argument of  $v$  is  $u$ -erased.

**Proof** The proposition is a corollary of Lemma 3.3.

**Definition 4.2** (1) We call a subterm  $s$  of a term  $t$  *unabsorbed in a reduction*  $P : t \rightarrow e$  if the descendants of  $s$  do not appear inside redex-arguments of terms in  $P$ , and call  $s$  *absorbed in*  $P$  otherwise. We call  $s$  *unabsorbed in*  $t$  if it is unabsorbed in any reduction starting from  $t$ , and *absorbed in*  $t$  otherwise.

(2) Let  $u_l$  be a redex in a term  $t$  defined as follows: choose an unabsorbed redex  $u_1$  in  $t$ ; choose an erased argument  $s_1$  of  $u_1$  that is not in normal form (if any); choose in  $s_1$  an unabsorbed redex  $u_2$ , and so on, as long as possible. Let  $u_1, s_1, u_2, \dots, u_l$  be such a sequence. Then we call  $u_l$  a *limit redex* and call  $u_1, s_1, u_2, \dots, u_l$  a *limit sequence* of  $t$ .

(3) We call a reduction *limit* if each contracted redex in it is limit, and call a strategy *limit* if in any term not in normal form it contracts a limit redex.

It is easy to show that a redex is unabsorbed iff it is *external* [5]. As in OTRSs [5, 7], it can be shown that in any term not in normal form there is an unabsorbed redex, hence a limit redex as well. The following lemma is also proved as in OTRSs [8].

**Lemma 4.2** Let  $u$  be a limit redex in  $t$  and let  $P : t \rightarrow e$ . Then there is no new redex in  $e$  that contains a descendant of  $u$  in its argument.

**Proof** Let  $u_1, s_1, u_2, \dots, u_l$  be the limit sequence of  $t$  with  $u_l = u$ . We prove by induction on  $|P|$  that (a): descendants of redexes  $u_1, \dots, u_l$  do not appear inside arguments of new redexes. If  $|P| = 0$ , then (a) is obvious. So let  $P : t \rightarrow e' \xrightarrow{v} e$ , let  $o$  be a descendant of  $u$  in  $e$ , and let  $o'$  be its ancestor in  $e'$ . It follows from the induction assumption that each redex  $u_i$  ( $i = 1, \dots, l-1$ ) has exactly one residual  $u'_i$  in  $e'$  (because contraction of a residual of any of the redexes  $u_1, \dots, u_{l-1}$  erases the descendant of  $u$ ), there is no new redex in  $e'$  that contains  $o'$  in its argument, and  $o$  is the only descendant of  $u$ . Thus if there is a new redex  $w$  in  $e$  that contains the residual  $u''_i$  of some  $u_i$  in its argument, then it must be created by  $v$ . If  $v \not\subseteq u'_1$ , then  $w$  contains  $u''_i$  in its argument iff

it contains the residual of  $u'_1$  in its argument, but this is impossible since  $u_1$  is unabsorbed. Thus  $v \subseteq u'_1$ . Let  $k$  be the maximal number such that  $v$  is in  $u'_k$  and let  $s'_k$  be the descendant of  $s_k$  in  $e'$ . Then  $v$  is in  $s'_k$  and contains  $u'_{k+1}$ . Let  $Q : s_k \rightarrow s''_k$  consist of steps of  $P$  that are made in descendants of  $s_k$ . Then the residual of  $u_{k+1}$  is in an argument of the new redex  $w \subseteq s''_k$ . But this is impossible since  $u_{k+1}$  is unabsorbed in  $s_k$ . Thus (a) is valid and the lemma is proved.

**Lemma 4.3** Let  $(\Sigma, R)$  be an OERS,  $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \dots \rightarrow t_n$  be a limit reduction in  $R$ , and  $P_\mu : s_0 = t_0 \xrightarrow{v_0} s_1 \xrightarrow{v_1} \dots \rightarrow s_n$  be its  $\mu$ -corresponding reduction in  $R_\mu$ . Then

- (1) for each  $k$  ( $0 \leq k \leq n$ ), the following holds:
  - (a)<sub>k</sub>:  $\|s_k\|_\mu = k$ ;
  - (b)<sub>k</sub>: each redex  $v'_k \subseteq s_k$  is  $\mu$ -main in  $s_k$ .
  - (c)<sub>k</sub>: in quasi-main arguments of any redex  $v''_k$  in  $s_k$  there are no  $\mu$ -symbols.
- (2) if  $P$  is normalizing, then so is  $P_\mu$ .

**Proof** (1) (a)<sub>0</sub> – (c)<sub>0</sub> are obvious. Suppose that (a)<sub>k</sub> – (c)<sub>k</sub> hold and let us show (a)<sub>k+1</sub> – (c)<sub>k+1</sub>. Let  $u_k = C[o_1, \dots, o_q]$  and  $v_k = C'[e_1, \dots, e_q, e'_1, \dots, e'_m]$ , where  $e_1, \dots, e_q$  are  $\mu$ -main arguments of  $v_k$  (which correspond to arguments  $o_1, \dots, o_q$  of  $u_k$ , respectively) and  $e'_1, \dots, e'_m$  are  $\mu$ -erased arguments of  $v_k$ . Since  $u_k$  and  $v_k$  are corresponding redexes in  $t_k$  and  $s_k$ , we have  $[v_k]_\mu = u_k$  and hence (α):  $[e_i]_\mu = o_i$  for all  $i = 1, \dots, q$ . Let  $e_{i_1}, \dots, e_{i_l}$  be the  $v_k$ -quasi-erased arguments and  $e_{j_1}, \dots, e_{j_p}$  be the  $v_k$ -quasi-main arguments. Then the contractum of  $v_k$  in  $R_\mu$  has the following form:  $o' = \mu\mu^0 \dots \mu^0 e'_1 \dots e'_m e_{i_1} \dots e_{i_l} o$ . By Proposition 4.1.(3),  $o_{i_1}, \dots, o_{i_l}$  are  $u_k$ -erased, and since  $u_k$  is limit, (β):  $o_{i_1}, \dots, o_{i_l}$  are in  $R$ -normal form. By (c)<sub>k</sub>, (γ): there are no occurrences of  $\mu$ -symbols in  $e_{j_1}, \dots, e_{j_p}, o$ . (Hence  $o$  coincides with the contractum of  $u_k$ ). It follows from (α), (β), and (b)<sub>k</sub> that (δ):  $e_{i_1}, \dots, e_{i_l}$  are in  $R_\mu$ -normal form.

By (γ),  $\|o'\|_\mu = \|v_k\|_\mu + 1$ . Hence  $\|s_{k+1}\|_\mu = \|s_k\|_\mu + 1 = k + 1$ , i.e., (a)<sub>k+1</sub> holds.

If  $v'_{k+1} \not\subseteq o'$ , then (b)<sub>k</sub> implies that  $v'_{k+1}$  is  $\mu$ -main. If  $v'_{k+1} \subseteq o'$ , then by (b)<sub>k</sub>,  $v'_{k+1} \not\subseteq e'_1, \dots, e'_m$  (since ancestors of  $e'_1, \dots, e'_m$  are  $\mu$ -erased arguments of  $v_k$ ) and by (δ),  $v'_{k+1} \not\subseteq e_{i_1}, \dots, e_{i_l}$ . Hence  $v'_{k+1} \subseteq o$  and by (γ),  $v'_{k+1}$  is  $\mu$ -main. Now (b)<sub>k+1</sub> is proved.

If  $o' \cap v''_{k+1} = \emptyset$ , then (c)<sub>k+1</sub> follows immediately from (c)<sub>k</sub>. If  $v''_{k+1} \subseteq o'$ , then as we have shown above (for  $v'_{k+1}$ ),  $v''_{k+1} \subseteq o$  and (c)<sub>k+1</sub> follows from (γ). Suppose now that  $o'$  is a proper subterm of  $v''_{k+1}$  and  $v''_{k+1}$  has an  $v_k$ -ancestor  $v^*_k$  in  $s_k$  for which  $v''_{k+1}$  is a residual. Let  $u^*_k$  be the corresponding redex of  $v^*_k$  in  $t_k$  (it exists because, by (b)<sub>k</sub>,  $v^*_k$  is  $\mu$ -main). Obviously,  $u_k$  is a proper subterm of  $u^*_k$  and since  $u_k$  is limit, it must be in an erased argument of  $u^*_k$ . By (c)<sub>k</sub>,  $v^*_k$ -quasi-main arguments do not contain  $\mu$ -symbols. Thus the sets of free variables of  $v^*_k$ -quasi-main arguments coincide with that of corresponding arguments of  $u^*_k$ . Hence, by Proposition 4.1.(2),  $v_k$  is in a quasi-erased argument of  $v^*_k$ . Therefore, by Proposition 4.1.(1),  $o'$  is in a quasi-erased argument of  $v''_{k+1}$  and the quasi-main arguments of  $v''_{k+1}$  coincide with the corresponding quasi-main arguments of  $v^*_k$ . Thus, by (c)<sub>k</sub>, in the quasi-main arguments of  $v''_{k+1}$  there are no occurrences of  $\mu$ -symbols. To prove (c)<sub>k+1</sub>, it remains to consider the case then  $o'$  is a proper subterm of  $v''_{k+1}$  and  $v''_{k+1}$  is created by  $v_k$ . If in quasi-main arguments of  $v''_{k+1}$  there are  $\mu$ -symbols, then in main arguments of corresponding redex  $u''_{k+1}$  in  $s_{k+1}$ , which is also an  $u_k$ -new redex, there are descendants of redexes contracted in  $P$ . But each redex contracted in  $P$  is a limit redex. Thus, by Lemma 4.2, their descendants can not occur in arguments of new redexes. Hence, also in this case, there are no  $\mu$ -symbols in quasi-main arguments of  $v''_{k+1}$ , and (c)<sub>k+1</sub> is valid.

- (2) By (b)<sub>n</sub>.

**Theorem 4.1** A limit strategy is perpetual in OERSs. Moreover, if a term  $t$  in an OERS  $R$  is strongly normalizable, then a limit strategy constructs a longest normalizing reduction starting from  $t$ , and its length coincides with the  $\mu$ -norm of  $R_\mu$ -normal form of  $t$ .

**Proof** If a limit  $R$ -reduction  $P$  starting from  $t$  is normalizing, then by Lemma 4.3 its corresponding  $R_\mu$ -reduction also is normalizing. Hence, by Lemma 4.1,  $t$  is strongly normalizable in  $R$ . Thus, the limit strategy is perpetual. Now, if  $t$  is strongly normalizable,  $Q$  is a normalizing  $R$ -reduction, and  $s$  is an  $R_\mu$ -normal form of  $t$ , then  $|Q| = |Q_\mu| \leq \|s\|_\mu = (\text{by Lemma 4.3}) = |P|$ .

A proof of perpetuality of the  $\beta$ -reduction strategy that contracts *special* redexes is presented in Barendregt [1]. It uses only unabsorbness of the leftmost-outermost redexes and therefore generalizes easily to the case of OERSs. The proof of the Conservation Theorem [2, 1] also remains valid for OERSs if one uses the limit strategy instead of the ‘special’ perpetual strategy (which does not work for OERSs). Therefore:

**Theorem 4.2 (Conservation Theorem)** If a term  $t$  in an OERS  $R$  has an infinite reduction and  $t \xrightarrow{u} s$ , where  $u$  is a non-erasing redex, then  $s$  has also an infinite reduction.

An extension of the Conservation Theorem can be found in de Groote [4]. It is a neat translation of Klop’s Lemma 4.1 by simulating  $\lambda_\mu$  with some reductions on  $\lambda$ -terms. Unfortunately, this is not possible for arbitrary OERSs.

As shown above (Lemmas 4.1 and 4.3), a term in an OERS  $R$  is strongly normalizable iff it is weakly normalizable in  $R_\mu$ . The following example shows that, despite the claim of Klop [10] (see p. 181), if  $R$  is strongly normalizing, then  $R_\mu$  need not be so.

**Example 4.1** Let  $R = \{f(\tau a(c, A)) \rightarrow g((\tau a(A, A)/a)A)\}$ , where  $c$  is a constant,  $a$  is an object metavariable, and  $\tau$  is a quantifier sign of arity  $(1, 2)$  and scope indicator  $(1, 2)$ . Creation or a redex inside the contractum is only possible if, say in the case  $a = x$ ,  $A$  has a subterm  $f(s)$ , i.e.,  $A = C[f(x)]$ , and  $\tau a(A, A) = \tau x(c, c)$ ; or if  $A = C[f(\tau yx)]$  with  $y \neq x$  and  $\tau a(A, A) = c$ , but this is impossible. Redex creation is not possible also outside the contractum, because  $g$  is not a pattern-symbol. Thus no redex creation is possible in  $R$  and hence  $R$  is strongly normalizing, while contraction of the redex  $v = f(\tau x(\mu^2(f(x), c), \mu^2(f(x), c)))$  in  $R_\mu$  creates itself:  $v \rightarrow \mu^2(f(x), g(\mu^2(v, c)))$ , and it is easy to see that  $v$  is not normalizable in  $R_\mu$ . (Recall that, for the case of OTRSs,  $R$  is strongly normalizing iff  $R_\mu$  is weakly normalizing [10, 8].)

## 5 Longest Reductions in Strongly Persistent OERSs

In this section, we design an algorithm for computing the lengths of longest reductions in *strongly persistent* ERSs. We restrict ourselves to the case of non-conditional ERSs, i.e., ERSs where an assignment is admissible iff the condition (a) of Definition 2.2 is satisfied. To this end, we introduce and study *strong similarity* of redexes. The results generalize some results from [8].

Without restricting the class of (non-conditional) OERSs, we can assume that in right-hand sides of rewrite rules the last argument of each metasubstitution is a term-metavariable or a metasubstitution.

**Definition 5.1** (1) Let  $t \xrightarrow{u} s$  in an OERS  $R$ , let  $t \rightarrow t' \twoheadrightarrow s$  be its refinement, and let  $v$  be a new redex in  $s$ . We call  $v$  *generated* if  $v$  is a residual of a redex of  $t'$  whose pattern is in the pattern of the contractum of  $u$  in  $R_f$ .

- (2) We call an OERS  $R$  *persistent* (written PERS) if each created redex in  $R$  is generated.  
(3) We call an OERS  $R$  *strongly persistent* (written SPERS) if  $R_{fS}$  is persistent.

It is easy to see that a non-simple OERS  $R$  is strongly persistent iff the left-hand sides of its rules contain only one function or quantifier symbol. Further, if  $R$  is a strongly persistent OERS, then so is  $R_\mu$ . Example 4.1 shows that if  $R$  is persistent, then  $R_\mu$  need not be so (in the reduction  $v = f(\tau x(\mu^2(f(x), c), \mu^2(f(x), c))) \xrightarrow{v} \mu^2(f(x), g(\mu^2(v, c)))$ , the pattern of the created  $v$  contains argument-symbols of the contracted  $v$ ). On the other hand, a PERS  $R$  such that  $R_\mu$  also is persistent need not be strongly persistent. An example is  $R = \{r_1 : \exists aA \rightarrow f((\tau aA/a)A), r_2 : g(f(x)) \rightarrow c\}$ . The proofs of results in this section use only persistency of  $R_\mu$  and hence are valid for a subclass of PERSs properly containing the class of SPERSs and persistent OTRSs (persistency of an OTRS  $R$  implies persistency of  $R_\mu$ ). We do not know a characterization of the class in terms of redex-creation.

**Notation** For any  $s \subseteq t$ ,  $MBV_R(s)$  (resp.  $MEBV_R(s)$ ) denotes the multiset of ( $R$ -essential) free occurrences of  $s$  bound by quantifiers belonging to patterns of  $R$ -redexes that are outside  $s$  (the number of occurrences of a bound variable in the multiset coincides with the number of its ( $R$ -essential) occurrences in  $s$ ).

**Definition 5.2** We call weakly similar redexes  $u = C[s_1, \dots, s_n]$  and  $v = C[o_1, \dots, o_n]$  respectively *similar*, *strongly similar*,  *$R$ -essentially similar*, or *strongly  $R$ -essentially similar* if, for all  $i$ ,  $BV(e_i) = BV(o_i)$ ,  $MBV(e_i) = MBV(o_i)$ ,  $EBV(e_i) = EBV(o_i)$ , or  $MEBV(e_i) = MEBV(o_i)$ . We denote the equivalence class of redexes strongly similar to  $u$  by  $\langle u \rangle_s$ .

**Lemma 5.1** Let  $R$  be an SPERS, let  $u = C[t_1, \dots, t_n]$  and  $v = C[s_1, \dots, s_n]$  be strongly similar  $R_\mu$ -redexes whose arguments are in normal form and are not variables. Further, let  $P : u = o_0 \xrightarrow{u} o_1 \xrightarrow{u_1} \dots$  and  $Q : v = e_0 \xrightarrow{v} e_1 \xrightarrow{v_1} \dots$  be refinements of rightmost  $R_\mu$ -reductions of  $u$  and  $v$  that are infinite or end at normal forms. Then it is possible to define one-to-one *correspondence* between the following occurrences of  $o_i$  and  $e_i$ :

- (1)  $\underline{S}$ -essential redexes and their arguments;
- (2)  $\underline{S}$ -essential descendants of redexes;
- (3)  $\underline{S}$ -essential descendants of arguments of  $u$  and  $v$ , called *argument subterms*; and
- (4)  $\underline{S}$ -essential descendants\* of free occurrences of variables in  $t_j$  and  $s_j$  that are bound in  $o_i$  and  $e_i$ , called *context variables*, where the notion of descendant\* is defined similarly to that of descendant with the exception that, during  $\underline{S}$ -steps, the occurrences of the replaced bound variables do *not* have descendants\*.

Furthermore, for each  $i$  ( $i = 0, 1, \dots$ ), the following conditions hold:

(a) <sub>$i$</sub> : corresponding  $\underline{S}$ -essential redexes in  $o_i$  and  $e_i$  are strongly  $\underline{S}$ -essentially similar, and  $u_i$  and  $v_i$  are corresponding  $\underline{S}$ -essential redexes if one of them is  $\underline{S}$ -essential.

(b) <sub>$i$</sub> : if  $o^*$  and  $e^*$ , as well as  $o''$  and  $e''$ , are corresponding  $\underline{S}$ -essential occurrences in  $o_i$  and  $e_i$ , then  $o^* \subseteq o''$  iff  $e^* \subseteq e''$ .

**Proof** By induction on  $i$  (the case  $i = 0$  is obvious from the assumptions). Assume first that  $u_m = \underline{S}y_1 \dots y_k t'_1 \dots t'_k t'_0$  and  $v_m = \underline{S}y_1 \dots y_k s'_1 \dots s'_k s'_0$  are  $\underline{S}$ -redexes. If  $u_m$  and  $v_m$  are  $\underline{S}$ -inessential, then all  $\underline{S}$ -essential occurrences of  $o_m$  and  $e_m$  are outside  $u_m$  and  $v_m$ , and each of them has exactly one  $\underline{S}$ -essential descendant in  $o_{m+1}$  and  $e_{m+1}$ , respectively. Hence descendants of corresponding  $\underline{S}$ -essential occurrences of  $o_m$  and  $e_m$  form pairs of corresponding  $\underline{S}$ -essential occurrences in  $o_{m+1}$  and  $e_{m+1}$ . So suppose that both  $u_m$  and  $v_m$  are  $\underline{S}$ -essential. It follows from (a) <sub>$m$</sub>  and (b) <sub>$m$</sub>  that

( $\alpha$ ):  $y_i$  has the same number of corresponding  $\underline{S}$ -essential occurrences in  $t'_0$  and  $s'_0$  and in each pair of corresponding  $\underline{S}$ -essential subterms of  $t'_0$  and  $s'_0$ . It follows from Lemma 3.1.(3) that descendants of  $\underline{S}$ -essential occurrences of  $o_m$  and  $e_m$  are  $\underline{S}$ -essential in  $o_{m+1}$  and  $e_{m+1}$  iff they are substituted for  $\underline{S}$ -essential context-variables. Thus corresponding  $\underline{S}$ -essential subterms in  $o_m$  and  $e_m$  have the same number of  $\underline{S}$ -essential descendants, and corresponding  $\underline{S}$ -essential context-variables have the same number of  $\underline{S}$ -essential descendants\* in  $o_{m+1}$  and  $e_{m+1}$ ; they form pairs of corresponding  $\underline{S}$ -essential occurrences in  $o_{m+1}$  and  $e_{m+1}$ . Since argument-subterms in  $o_m$  and  $e_m$  are not variables, different subterms have different descendants. Thus the correspondence between these subterms in  $o_{m+1}$  and  $e_{m+1}$  remains one-to-one. Since  $R_\mu$  is persistent, no new redexes are created by these steps. Thus  $(a)_{m+1}$  follows from  $(a)_m$  and from the fact that the context-variables form pairs of corresponding occurrences.  $(b)_{m+1}$  follows from  $(\alpha)$  and  $(b)_m$ .

Suppose now that  $u_m$  and  $v_m$  are  $R_{\mu f}$ -redexes. In this case, there are no  $\underline{S}$ -redexes in  $o_m$  and  $e_m$ . Obviously, the contractum of  $u_m$  can be obtained from the contractum of  $v_m$  by replacing descendants of arguments of  $v_m$  with the corresponding arguments of  $u_m$ . Since  $R_\mu$  is persistent, for each new redex  $w$  in  $o_{m+1}$  there is a unique new redex  $w'$  in  $e_{m+1}$ . All the descendants of the occurrences that are outside  $u_m$  and  $v_m$  are  $\underline{S}$ -essential. By  $(a)_m$ ,  $u_m$  and  $v_m$  are strongly similar. Hence, it follows from conditions (a)-(b) of Definition 2.2 and Lemma 3.2 that corresponding new redexes in  $o_{m+1}$  and  $e_{m+1}$  are either both  $\underline{S}$ -essential or both are  $\underline{S}$ -inessential. The same holds for corresponding arguments of the corresponding redexes. Corresponding occurrences in  $o_m$  and  $e_m$  have the same number of  $\underline{S}$ -essential descendants in  $o_{m+1}$  and  $e_{m+1}$ ; together with corresponding  $\underline{S}$ -essential new redexes they form pairs of corresponding  $\underline{S}$ -essential occurrences in  $o_{m+1}$  and  $e_{m+1}$ ; the correspondence remains one-to-one. Since variables bound by quantifiers belonging to patterns of  $w$  and  $w'$  can only occur in the descendants of arguments of  $u_m$  and  $v_m$ , and  $\underline{S}$ -essential (in  $o_{m+1}$  and  $e_{m+1}$ ) occurrences of context variables form pairs of corresponding  $\underline{S}$ -essential occurrences in corresponding arguments of  $w$  and  $w'$ , it follows from Lemma 3.1.(3) that  $w$  and  $w'$  are strongly  $\underline{S}$ -essentially similar. Hence  $(a)_{m+1}$  follows from  $(a)_m$ .  $(b)_{m+1}$  follows from  $(b)_m$ .

**Definition 5.3** Let  $R$  be an SPTRS.

(1) Let  $t$  be a term in  $R_\mu$ , let  $s$  be a non-variable subterm of  $t$ , and let  $P : t \rightarrow e$  be the rightmost innermost normalizing  $R_\mu$ -reduction. Then we define  $Mult_\mu(s, t)$  as the number of  $P$ -descendants of  $s$  in  $e$ .

(2) Let  $u = C[e_1, \dots, e_n]$  be a redex in  $R_\mu$ , let  $s' \subseteq e_i$ , let  $v = C[o_1, \dots, o_n]$  be a redex strongly similar to  $u$  whose arguments  $o_1, \dots, o_n$  are in  $R_\mu$ -normal form and are not variables, and let  $Q : v \rightarrow o$  be the rightmost innermost normalizing  $R_\mu$ -reduction. Then we define  $mult_\mu(u, i) = mult_\mu(u, s') = mult_\mu(\langle u \rangle_s, i) = Mult_\mu(o_i, v)$ .  $mult_\mu(u) = mult_\mu(\langle u \rangle_s)$  is defined to be the number of  $\mu$ -subterms in  $o$  that appear during  $Q$ , i.e., that are not descendants of  $\mu$ -subterms from  $v$ . We call numbers  $mult_\mu(u, i)$  and  $mult_\mu(\langle u \rangle_s, i)$  *proper  $\mu$ -indices* of  $u$  and  $\langle u \rangle_s$ , and numbers  $mult_\mu(u)$  and  $mult_\mu(\langle u \rangle_s)$   *$\mu$ -indices* of  $u$  and  $\langle u \rangle_s$ .

The correctness of the above definition follows from Lemma 5.1. We have to consider only non-variable subterms because descendants of a bound variable with different indexes can happen to coincide. Alternatively, one could consider all subterms but take into account the multiplicity of each descendant.

The following lemma is proved using Definition 5.3 and the fact that if  $t$  be a strongly normalizable term in a PERS  $R_\mu$ ,  $e \subseteq s \subseteq t$ , and  $e$  and  $s$  are non-variable  $R_\mu$ -normal forms, then  $Mult_\mu(s, t) = Mult_\mu(e, t)$ .

Below  $L(t)$  denotes the length of a longest reduction starting from  $t$ .

**Lemma 5.2** (1) Let  $t$  be a strongly normalizable term in an SPERS  $R$  and  $u_1, \dots, u_n$  be all redexes in  $t$ . Then

$$L(t) = \sum_{i=1}^n Mult_{\mu}(u_i, t) mult_{\mu}(u_i).$$

(2) Let  $t$  be a strongly normalizable term in an PERS  $R_{\mu}$  and  $u_1, \dots, u_n$  be all redexes in  $t$  that contain a non-variable subterm  $s$  in their arguments. Suppose that  $s$  is in  $m_i$ -th argument of  $u_i$  ( $i = 1, \dots, n$ ). Then

$$Mult_{\mu}(s, t) = \prod_{i=1}^n mult_{\mu}(u_i, s) = \prod_{i=1}^n mult_{\mu}(u_i, m_i).$$

(3) Let  $u = C[e_1, \dots, e_k]$  be a redex whose arguments  $e_1, \dots, e_k$  are not variables and are in normal form, in an SPERS  $R$ . Then, for all  $j = 1, \dots, k$ ,

$$mult_{\mu}(u, j) = mult_{\mu}(\langle u \rangle_s, j) = \sum_{i=1}^{m_j} Mult_{\mu}(e_{j_i}, o),$$

$$mult_{\mu}(u) = mult_{\mu}(\langle u \rangle_s) = \sum_{i=1}^m Mult_{\mu}(u_i, o) mult_{\mu}(u_i) + 1,$$

where  $o$  is the contraction of  $u$  in  $R_{\mu}$ ,  $e_{j_1}, \dots, e_{j_{m_j}}$  are all descendants of  $e_j$  in  $o$ , and  $u_1, \dots, u_m$  are all redexes in  $o$ .

**Lemma 5.3** Let  $u$  and  $v$  be strongly similar redexes in an SPERS  $R$ , let  $u \xrightarrow{u} o$  and  $v \xrightarrow{v} e$ . Then  $u$  and  $v$  create the same number of strongly similar redexes.

**Proof** Take in Lemma 5.1 the refinement of  $u$  for  $P$  and the refinement of  $v$  for  $Q$ .

Hence, one can define  $\langle u_0 \rangle_s, \langle u_1 \rangle_s \dots$  to be a  $\langle u_0 \rangle_s$ -chain or  $u_0$ -chain if  $\langle u_i \rangle_s$  generates  $\langle u_{i+1} \rangle_s$ , i.e., if  $u_i$  generates a redex strongly similar to  $u_{i+1}$ . It is easy to see that a term  $t$  in a PERS  $R$  is strongly normalizable iff all chains of redexes in  $t$  are finite ( $(\Rightarrow)$  is trivial and  $(\Leftarrow)$  can be shown using the method of multiset ordering; we could equally use similarity classes for this criterion instead of strong similarity classes). Therefore, the following theorem is a corollary of Theorem 4.1 and Lemma 5.2.

**Theorem 5.1** Let  $t$  be a term in an SPERS  $R$ . Then the least upper bound  $L(t)$  of lengths of reductions starting from  $t$  can be found using the following:

**Algorithm 5.1** Let  $\langle u_1 \rangle_s, \dots, \langle u_n \rangle_s$  be all strong similarity classes whose member redexes occur in  $t$ . If an  $\langle u_i \rangle_s$ -chain is infinite for at least one  $i$ , then  $L(t) = \infty$ . Otherwise, using Lemmas 5.2.(3) and 5.2.(2), compute the  $\mu$ -indices and the proper  $\mu$ -indices of all classes  $\langle u_i \rangle_s$ . Finally, using Lemmas 5.2.(2) and 5.2.(1), compute  $L(t)$ .

For any  $\langle u \rangle_s$  in a non-creating OERS (where no redex-creation is possible),  $mult_{\mu}(\langle u \rangle_s) = 1$ ,  $mult_{\mu}(\langle u \rangle_s, i) = 1$  if the  $i$ -th argument  $o_i$  of  $u$  does not have a  $u$ -descendant, and  $mult_{\mu}(\langle u \rangle_s, i)$  coincides with the number of  $u$ -descendants of  $o_i$  otherwise. Since developments in an OERS are reductions in non-creating OERSs up to some isomorphism, a simpler form of Algorithm 5.1 can be used for computing lengths of longest developments in OERSs.

## 6 Conclusions

We presented an algorithm for constructing longest reductions and computing their lengths in OERSs, and used the algorithm to generalize the Conservation Theorem to OERSs. Bergstra and Klop [3] gave a characterization of erasing  $\beta$ -redexes ( $K$ -redexes) for which the Conservation Theorem in  $\lambda$ -calculus still is valid. We leave this question for OERSs to a future investigation. De Vrijer [17] gave another characterization of lengths of longest reductions in a typed  $\lambda$ -calculus. Van de Pol [16] used a semantic method for strong normalization proofs in higher order rewriting systems.

**Acknowledgments** I enjoyed discussions with H. Barendregt, J. W. Klop, J.-J. Lévy, L. Maranget, G. Mints, V. van Oostrom, Sh. Pkhakadze, F. van Raamsdonk, Kh. Rukhaia, and V. Sazonov. I also would like to thank G. Gonthier, G. Tagviashvili, K. Urbaitis, and F. J. de Vries for their help in the preparation of this paper.

## References

- [1] Barendregt H. P. The Lambda Calculus, its Syntax and Semantics. North-Holland, 1984.
- [2] Barendregt H. P., Bergstra J., Klop J. W., Volken H. Some notes on lambda-reduction, In: Degrees, reductions, and representability in the lambda calculus. Preprint no. 22, University of Utrecht, Department of mathematics, p. 13-53, 1976.
- [3] Bergstra J. A., Klop J. W. Strong normalization and perpetual reductions in the Lambda Calculus. J. of Information Processing and Cybernetics 18, 1982, p. 403-417.
- [4] De Groote P. The Conservation Theorem revisited. In: M. Bezem, J. F. Groote, eds. Proc. of the International Conference on Typed Lambda Calculi and Applications. Springer LNCS, vol. 664, Utrecht, 1993, p. 163-178.
- [5] Huet G., Lévy J.-J. Computations in Orthogonal Rewriting Systems. In: J.-L. Lassez and G. Plotkin, eds. Computational Logic, Essays in Honor of Alan Robinson. MIT Press, 1991, p. 395-443.
- [6] Khasidashvili Z. The Church-Rosser theorem in Orthogonal Combinatory Reduction Systems. Report 1825, INRIA Rocquencourt, 1992.
- [7] Khasidashvili Z. Optimal normalization in orthogonal term rewriting systems. In: C. Kirchner, ed. Proc. of 5<sup>th</sup> International Conference on Rewriting Techniques and Applications. Springer LNCS, vol. 690, Montreal, 1993, p. 243-258.
- [8] Khasidashvili Z. Perpetuality and strong normalization in orthogonal term rewriting systems. In: P. Enjalbert, E. W. Mayr, and K. W. Wagner eds. Proc. of 11<sup>th</sup> Symposium on Theoretical Aspects of Computer Science. Springer LNCS, vol. 775, Caen, 1994, p. 163-174.
- [9] Khasidashvili Z. Perpetual reductions in orthogonal combinatory reduction systems. CWI report CS-R9349, 1993.

- [10] Klop J. W. Combinatory Reduction Systems. Mathematical Centre Tracts n. 127, CWI, Amsterdam, 1980.
- [11] Klop J. W. Term Rewriting Systems. In: S. Abramsky, D. Gabbay, and T. Maibaum eds. Handbook of Logic in Computer Science, vol. II, Oxford University Press, 1992, p. 1-116.
- [12] Klop J. W., van Oostrom V., van Raamsdonk F. Combinatory reduction Systems: introduction and survey. In: To Corrado Böhm, J. of Theoretical Computer Science 121, 1993, p. 279-308.
- [13] Maranget L. La stratégie paresseuse, Thèse de l'Université de Paris VII, 1992.
- [14] Nederpelt R. P. Strong Normalization for a typed lambda-calculus with lambda structured types. Ph.D. Thesis, Eindhoven, 1973.
- [15] O'Donnell M. J. Computing in systems described by equations. Springer LNCS 58, 1977.
- [16] Van de Pol J. Termination proofs for Higher-order Rewrite Systems. To appear in: Proc. of International Workshop on Higher-order Algebra, Logic and Term Rewriting, Amsterdam, 1993. Available as logic group preprint series No. 93, Utrecht, 1993.
- [17] De Vrijer R. C. Surjective Pairing and Strong Normalization: Two Themes in Lambda Calculus. Ph.D. Thesis, Amsterdam, 1987.