Perpetuality and Uniform Normalization

Zurab Khasidashvili and Mizuhito Ogawa

NTT Basic Research Laboratories
3-1 Morinosato-Wakamiya, Atsugi, Kanagawa, 243-01, Japan
{zurab,mizuhito}@theory.brl.ntt.co.jp

Abstract. We define a perpetual one-step reduction strategy which enables one to construct *minimal* (w.r.t. Lévy's ordering $\trianglelefteq$ on reductions) infinite reductions in Conditional Orthogonal Expression Reduction Systems. We use this strategy to derive two characterizations of *perpetual* redexes, i.e., redexes whose contractions retain the existence of infinite reductions. These characterizations generalize existing related criteria for perpetuality of redexes. We give a number of applications of our results, demonstrating their usefulness. In particular, we prove equivalence of weak and strong normalization (the *uniform normalization* property) for various restricted λ-calculi, which cannot be derived from previously known perpetuality criteria.

1 Introduction

The objective of this paper is to study sufficient conditions for *uniform normalization*, UN, of a term in an orthogonal (first or higher-order) rewrite system, and for the UN property of the rewrite system itself. Here a term is UN if either it does not have a normal form, or if any reduction eventually terminates in a normal form; the rewrite system is UN if every term is UN. Interest in criteria for UN arises, for example, in the proofs of strong normalization of typed λ-calculi, as it relates to the work on reducing strong normalization proofs to proving weak normalization [Ned73, Klo80, dVr87, dGr93, Kha94c]. Further, the question: 'Which classes of terms have the uniform normalization property?' is posed in [BI94] in connection with finding UN solutions to fixed point equations, and with representability of partial recursive functions by UN-terms only, in the λ-calculus.[1] The UN property is clearly useful as then all strategies are normalizing, and in particular, there is more room for optimality (cf. [GK96]).

It is easy to see that a rewriting system is UN iff all of its redexes are *perpetual.* These are redexes that reduce terms having an infinite reduction, which we call ∞-terms, to ∞-terms. Therefore, studying the UN property reduces to studying perpetuality of redexes. The latter has already been studied quite extensively in the literature. The classical results in this direction are *Church's Theorem* [CR36], stating that the λ_I-calculus is uniformly normalizing, and the *Conservation Theorem* of Barendregt et al [BBKV76, Bar84], stating that β_I-redexes are perpetual in the λ-calculus. Bergstra and Klop [BK82] give a sufficient and necessary criterion for perpetuality of β_K-redexes in every context. Klop [Klo80] generalized Church's Theorem to all non-erasing orthogonal Combinatory Reduction Systems (CRSs) by

[1] The UN property is called *strong normalization* in [BI94].

showing that the latter are UN, and Khasidashvili [Kha94c] generalized the Conservation Theorem to all orthogonal Expression Reduction Systems (ERSs) [Kha92], by proving that all non-erasing redexes are perpetual in orthogonal ERSs.

For orthogonal Term Rewriting Systems (OTRSs), a very powerful perpetuality criterion was obtained by Klop [Klo92] in terms of *critical* redexes. These are redexes that are not perpetual, i.e., reduce ∞-terms to strongly normalizable terms (*SN-terms*). Klop showed that any critical redex u must erase an argument possessing an infinite reduction. The later is not true for higher-order rewrite systems, because substitutions (from the outside) into the arguments of u may occur during rewrite steps, which may turn an SN argument of u into an ∞-term. However, we show that a critical redex u in a term t must necessarily erase a *potentially infinite* argument, i.e., an argument that would become an ∞-(sub)term after a number of (*passive*) steps in t. From this, we derive a criterion, called *safety*, of perpetuality of redexes in every context, similar to the perpetuality criterion of β_K-redexes [BK82]. These are the main results of this paper, and we will demonstrate their usefulness in applications.

We obtain our results in the framework of *Orthogonal (Context-sensitive) Conditional Expression reduction Systems* (OCERSs) [KO95]. CERS is a format for higher-order rewriting, or to be precise, second-order rewriting, which extends ERSs [Kha92] by allowing restrictions both on arguments of redexes and on the contexts in which the redexes can be contracted. Various interesting typed λ-calculi, including the simply typed λ-calculus and the system **F** [Bar92], can directly be encoded as OCERSs (see also [KOR93]); λ–calculi with specific reduction strategies (such as the call-by-value λ-calculus [Plo75]) can also be naturally encoded as OCERSs. ERSs are very close to the more familiar format of CRSs of Klop [Klo80], and we claim that all our results are valid for orthogonal CRSs as well (see [Raa96] for a detailed comparison of various forms of higher-order rewriting). However, using an example due to van Oostrom [Oos97], we will demonstrate that our results cannot be extended to higher-order rewriting systems where function variables can be bound [Wol93, Nip93, OR94], as they can exhibit pretty strange behaviour not characteristic of the λ-calculi.

In order to prove our perpetuality criteria, we first generalize the *constricting* (or *zoom-in*) perpetual strategy, independently discovered by Plaisted [Pla93], Sørensen [Sør95], Gramlich [Gra96], and Melliès [Mel96] (with small differences), from term rewriting and the λ-calculus to OCERSs. These strategies specify a construction of infinite reductions (whenever possible) such that all steps are performed in some smallest ∞-subterm. Our strategy is slightly more general than the above, and can be restricted so that the computation becomes constricting, and this allows for simple and concise proofs of our perpetuality criteria. We also show that constricting perpetual reductions are minimal w.r.t. Lévy's ordering on reductions in orthogonal rewriting systems [Lév80, HL91].

Despite the fact that our criteria are simple and intuitive, they appear to be strong tools in proving strong normalization from weak normalization in orthogonal (typed or type-free) rewrite systems. We will show that previously known related criteria [CR36, BBKV76, BK82, Klo80, Klo92, Kha94c] can be obtained as special cases. We will also derive the UN property for a number of variations of β-reductions [Plo75, dGr93, BI94, HL93], which cannot be derived from previously known perpetuality criteria, as immediate consequences of our criteria.

2 Conditional Expression Reduction Systems

In this section, we recall the basic theory of orthogonal Conditional Expression Reduction Systems, OCERSs, as developed in [KO95], and some results concerning *similarity* of redexes in OERSs from [Kha94c]. CERSs extend *Expression Reduction Systems* [Kha92], a formalism of higher-order (rather, second-order) rewriting close to *Combinatory Reduction Systems* [Klo80]. We refer to [Raa96] for an extensive survey of the relationship between various formats of higher-order rewriting, such as [Klo80, Kha92, Wol93, Nip93, OR94]. Restricted rewriting systems with substitutions were first studied in [Pkh77] and [Acz78]. We refer to [Klo92] for a survey of results concerning conditional TRSs.

Terms in CERSs are built from the alphabet like in the first order case. The symbols having binding power (like λ in λ-calculus or $\int$ in integrals) require some binding variables and terms as arguments, as specified by their arity. Scope indicators are used to specify which variables have binding power in which arguments. For example, a β-redex in the λ-calculus appears as $Ap(\lambda x\, t, s)$, where Ap is a function symbol of arity 2, and λ is an operator sign of arity $(1,1)$ and scope indicator (1). Integrals such as $\int_s^t f(x)\,dx$ can be represented as $\int x(s,t,f(x))$ using an operator sign $\int$ of arity $(1,3)$ and scope indicator (3).

Metaterms will be used to write rewrite rules. They are constructed from metavariables and meta-expressions for substitutions, called metasubstitutions. Instantiation of metavariables in metaterms yields terms. Metavariables play the role of variables in the TRS rules, and function variables in HRS and HORS rules [Nip93, OR94]. Differently from HRSs and HORSs, metavariables *cannot* be bound.

Definition 2.1 Let Σ be an *alphabet* comprising *variables*, denoted by x, y, z and *symbols* (*signs*). A symbol σ can be either a *function symbol* (*simple operator*) having an *arity* $n \in N$, or an *operator sign* (*quantifier sign*) having *arity* $(m,n) \in N \times N$. In the latter case σ needs to be supplied with m *binding* variables $x_1,\ldots,x_m$ to form the *quantifier* (*compound operator*) $\sigma x_1 \ldots x_m$. If σ is an operator sign it also has a *scope indicator* specifying, for each variable, in which of the n arguments it has binding power. *Terms* t, s, e, o are constructed from variables, function symbols and quantifiers in the usual first-order way, respecting (the second component of the) arities. A predicate AT on terms specifies which terms are *admissible*.

Metaterms are constructed like terms, but also allowing as basic constructions *metavariables* A, B, ... and *metasubstitutions* $(t_1/x_1,\ldots,t_n/x_n)t_0$, where each t_i is an arbitrary metaterm and the x_i have a binding effect in t_0. An *assignment* θ maps each metavariable to some term. The application of θ to a metaterm t is written $t\theta$ and is obtained from t by replacing metavariables with their values under θ, and by replacing metasubstitutions $(t_1/x_1,\ldots,t_n/x_n)t_0$, in right to left order, with the result of substitution of terms $t_1,\ldots,t_n$ for free occurrences of $x_1,\ldots,x_n$ in t_0.

The specification of a CERS consists of an alphabet (generating a set of terms possibly restricted by the predicate AT) as specified above and a set of rules (generating the rewrite relation possibly restricted by the predicates AA and AC) as specified below. The predicate AT can be used to express sorting and typing constraints, since sets of admissible terms allowed for arguments of an operator can be seen as terms of certain sorts or types.

The ERS syntax is very close to the syntax of the λ-calculus. For example, the β-rule is written as $Ap(\lambda xA, B) \rightarrow (B/x)A$, where A and B can be instantiated by any terms. The η-rule is written as $\lambda xAp(A, x) \rightarrow A$, where it is required that $x \notin A\theta$ for an assignment θ, otherwise an x occurring in $A\theta$ and therefore bound in $\lambda x(A\theta, x)$ would become free. A rule like $f(A) \rightarrow \exists x(A)$ is also allowed, but in that case the assignment θ with $x \in A\theta$ is not. Such a collision between free and bound variables cannot arise for restricted (by the condition $(*)$ below, see Definition 2.2) assignments.

Definition 2.2 A rewrite rule is a (named) pair of metaterms $r : t \rightarrow s$, such that t and s do not contain free variables. We close the rules under assignments: $r\theta : t\theta \rightarrow s\theta$ if $r : t \rightarrow s$ and θ is an assignment. To avoid the capturing of free variables, this is restricted to assignments θ such that

$(*)$ *each free variable occurring in a term $A\theta$ assigned to a metavariable A is either bound in the θ-instance of each occurrence of A in the rule or in none of them.*

The term $t\theta$ is then called a *redex* and $s\theta$ its *contractum*. We close under contexts $C[r\theta] : C[t\theta] \rightarrow C[s\theta]$, if $r\theta : t\theta \rightarrow s\theta$ and $C[\,]$ is a context (a term with one hole).

The rewrite relation thus obtained is the usual (unconditional, context-free) ERS-rewrite relation. If restrictions are put on assignments, via an *admissibility* predicate AA on rules and assignments, the rewrite relation will be called *conditional*. We call redexes that are instances of the same rule (i.e., with the same admissibility predicate) *weakly similar*. If restrictions are put on contexts, via a predicate AC on rules, assignments and contexts, the rewrite relation will be called *context-sensitive*.

A *CERS* R is a pair consisting of an alphabet and a set of rewrite rules, both possibly restricted.

Note that we allow metavariable-rules like $\eta^{-1} : A \rightarrow \lambda xAp(Ax)$ and metavariable-introduction-rules like $f(A) \rightarrow g(A, B)$, which are usually excluded a priori. This is only useful when the system is conditional.

Let $r : t \rightarrow s$ be a rule in a CERS R and let θ be admissible for r. Subterms of a redex $v = t\theta$ that correspond to the metavariables in t are the *arguments* of v, and the rest is the *pattern* of v (hence the binding variables of the quantifiers occurring in the pattern belong to the pattern too). Subterms of v rooted in the pattern are called the *pattern-subterms* of v.

Notation We use a, b, c, d for constants, t, s, e, o for terms and metaterms, u, v, w for redexes, and N, P, Q for reductions. We write $s \subseteq t$ if s is a subterm of t. A one-step reduction in which a redex $u \subseteq t$ is contracted is written as $t \xrightarrow{u} s$ or $t \rightarrow s$ or just u. We write $P : t \twoheadrightarrow s$ or $t \overset{P}{\twoheadrightarrow} s$ if P denotes a reduction (sequence) from t to s, write $P : t \twoheadrightarrow$ if P may be infinite, and write $P : t \twoheadrightarrow \infty$ if P is infinite (i.e, of the length ω). $P + Q$ denotes the concatenation of P and Q. $FV(t)$ denotes the set of free (i.e., unbound) variables of t.

Below, when we speak about terms and redexes, we will always mean admissible terms and admissible redexes, respectively.

2.1 Orthogonal CERSs

The idea of orthogonality is that contraction of a redex does not destroy other redexes (in whatever way), but rather leaves a number of their residuals. A prerequisite for the definition of residual is the notion of *descendant*, also called *trace*, allowing tracing of subterms during a reduction. Whereas this is pretty simple in the first-order case, ERSs may exhibit complex behaviour due to the possibility of nested metasubstitutions, thereby complicating the definition of descendants. However, it is a standard technique in higher-order rewriting [Klo80] to *decompose* or *refine* each rewrite step into two parts: a *TRS*-part replacing the left-hand side by the right-hand side without evaluating the (meta)substitutions and a *substitution*-part evaluating the delayed substitutions. To express substitution, we use the S-reduction rules

$$S^{n+1}x_1 \dots x_n A_1 \dots A_n A_0 \to (A_1/x_1, \dots, A_n/x_n)A_0, \quad n = 1, 2, \dots,$$

where S^{n+1} is the *operator sign of substitution* with arity $(n, n+1)$ and scope indicator $(n+1)$, and $x_1, \dots, x_n$ and $A_1, \dots, A_n, A_0$ are pairwise distinct variables and metavariables.[2] Thus S^{n+1} binds only in the last argument. The difference with β-rules is that S-reductions can only perform β-developments of λ-terms, so one can think of them as (simultaneous) let-expressions.

Thus the descendant relation of a rewrite step can be obtained by composing the descendant relation of the TRS-step and the descendant relations of the S-reduction steps. All known concepts of descendants agree in the cases when the subterm $s \subseteq t$ which is to be traced during a step $t \xrightarrow{u} o$ is in an argument of the contracted redex u, properly contains u, or does not overlap with it. The differences occur in the case when s is a pattern-subterm, in which case we define the contractum of u to be the descendant of s, while according to many (especially early) definitions, s does not have a u-descendant.

We will explain the concept with examples. Consider first a TRS-step $t = f(g(a)) \to b = s$ performed according to the rule $f(g(x)) \to b$. The descendant of both pattern-subterms $f(g(a))$ and $g(a)$ of t in s is b, and a does not have a descendant in s. The refinement of a β-step $t = Ap(\lambda x(Ap(x,x)), z) \to_\beta f(z) = e$ would be $t = Ap(\lambda x(Ap(x,x)), z) \to_{\beta_f} o = S^2 xzAp(x,x) \to_S f(z) = e$; the descendant of both t and $\lambda x(Ap(x,x))$ after the *TRS*-step is the contractum $S^2 xzAp(x,x)$, and the descendants of $Ap(x,x), z \subseteq t$ are respectively the subterms $Ap(x,x), z \subseteq o$; the descendant of both $o = S^2 xzAp(x,x)$ and $Ap(x,x)$ after the substitution step is the contractum e; the descendant of $z \subseteq o$, as well as of the bound occurrence of x in $Ap(x,x)$, is the occurrence of z in e.

Definition 2.3 Let $t \xrightarrow{u} s$ in a CERS R, let $v \subseteq t$ be an admissible redex, and let $w \in s$ be a u-descendant of v. We call w a *u-residual* of v if (a) the patterns of u and v do not overlap; (b) w is a redex weakly similar to v; and (c) w is admissible. (So u itself does not have *u-residuals* in s.) The notion of *residual* of redexes extends naturally to arbitrary reductions. A redex in s is called a *new* redex or a *created* redex if it is not a residual of a redex in t. The *ancestor* relation is converse to that of descendant, and the *predecessor* relation to that of residual.

[2] We assume that the CERS does not contain the symbols S^{n+1}.

Definition 2.4 ([KO95]) A CERS is called *orthogonal* (OCERS) if:

- the left-hand side of a rule is not a single metavariable,
- the left-hand side of a rule does not contain metasubstitutions, and its metavariables contain those of the right-hand side,
- all the descendants of an admissible redex u in a term t under the contraction of any other admissible redex $v \subseteq t$ are residuals of u.

The second condition ensures that rules exhibit deterministic behaviour when they can be applied. The last condition is the counterpart of the *subject reduction property* in typed λ-calculi [Bar92]. For example, consider the rules $a \rightarrow b$ and $f(A) \rightarrow A$ with admissible assignment $A\theta = a$. The descendant $f(b)$ of the redex $f(a)$ after contraction of a is not a redex since the assignment $A\theta = b$ is not admissible, hence the system is not orthogonal.

As in the case of the λ-calculus [Bar84], for any co-initial (i.e., with the same initial term) reductions P and Q, one can define in OCERSs the notion of *residual of P under Q*, written P/Q, due to Lévy [Lév80]. We write $P \trianglelefteq Q$ if $P/Q = \emptyset$ ($\trianglelefteq$ is the *Lévy-embedding* relation); P and Q are called *Lévy-equivalent*, *strongly-equivalent*, or *permutation-equivalent* (written $P \approx_L Q$) if $P \trianglelefteq Q$ and $Q \trianglelefteq P$. It follows easily from the definition of / that, for any (appropriate) P' and Q', $(P + P')/Q \approx_L P/Q + P'/(Q/P)$ and $P/(Q + Q') \approx_L (P/Q)/Q'$.

Theorem 2.5 (Strong Church-Rosser [KO95]) For any finite co-initial reductions P and Q in an OCERS, $P + (Q/P) \approx_L Q + (P/Q)$.

2.2 Similarity of redexes

The idea of *similarity* of redexes [Kha94] u and v is that u and v are weakly similar, i.e., match the same rewrite rule, and quantifiers in the pattern of u and v bind 'similarly' in the corresponding arguments. Consequently, for any pair of corresponding arguments of u and v, either both are erased after contraction of u and v, or none is. For example, recall that a β-redex $Ap(\lambda xt, s)$ is an *I-redex* if $x \in FV(t)$, and is a *K-redex* otherwise. Then, all I-redexes are similar, and so are all K-redexes, but no I-redex is similar to a K-redex.

We can write a CERS redex as $u = C[\overline{x_1}t_1, \ldots, \overline{x_n}t_n]$, where C is the pattern, $t_1, \ldots, t_n$ are the arguments, and $\overline{x_i} = \{x_{i_1}, \ldots, x_{i_{n_i}}\}$ is the subset of binding variables of C such that t_i is in the scope of an occurrence of each x_{i_j}, $i = 1, \ldots, n$. Let us call the maximal subsequence $j_1, \ldots, j_k$ of $1, \ldots, n$ such that $t_{j_1}, \ldots, t_{j_k}$ have u-descendants the *main sequence* of u or the *u-main sequence*, call $t_{j_1}, \ldots, t_{j_k}$ (u-)*main arguments*, and call the remaining arguments (u)-*erased*. Now the similarity of redexes can be defined as follows:

Definition 2.6 Let $u = C[\overline{x_1}t_1, \ldots, \overline{x_n}t_n]$ and $v = C[\overline{x_1}s_1, \ldots, \overline{x_n}s_n]$ be weakly similar. We call u and v *similar*, written $u \sim v$, if the main sequences of u and v coincide, and for any main argument t_i of u, $\overline{x_i} \cap FV(t_i) = \overline{x_i} \cap FV(s_i)$.

The following lemma, whose proof is similar to that of Lemma 3.3 in [Kha94c], shows that only pattern-bindings (i.e., bindings from inside the pattern) of free variables in *main* arguments of a redex are relevant for the erasure of its arguments. Below, θ will (besides denoting assignments) also denote substitutions assigning terms to variables; when we write $o' = o\theta$ for some substitution θ, we assume that no free variables of the substituted subterms become bound in o' (i.e., we rename bound variables in o when necessary).

Lemma 2.7 Let $u = C[\overline{x_1}t_1, \ldots, \overline{x_n}t_n]$ and $v = C[\overline{x_1}s_1, \ldots, \overline{x_n}s_n]$ be weakly similar redexes, and let for any main argument t_i of u, $\overline{x_i} \cap FV(t_i) = \overline{x_i} \cap FV(s_i)$. Then the main sequences of u and v coincide, and consequently, $u \sim v$. In particular, if $u = v\theta$, then $u \sim v$.

3 A Minimal Perpetual Strategy

In this section we generalize the constricting perpetual strategy [Pla93, Sør95, Gra96, Mel96] from TRSs and the λ-calculus to all OCERSs.

Let us first fix the terminology. Recall that a term t is called *weakly normalizable*, a *WN-term*, written $WN(t)$, if it is reducible to a *normal form*, i.e., a term without a redex. t is called *strongly normalizable*, an *SN-term*, written $SN(t)$, if it does not possess an infinite reduction. We call t an ∞-*term*, $\infty(t)$, if $\neg SN(t)$. Clearly, for any term t, $SN(t) \Rightarrow WN(t)$. If the converse is also true, then we call t *uniformly normalizable*, or a *UN-term*. So for UN-terms t, either t does not have a normal form, or all reductions from t eventually terminate. Correspondingly, a rewrite system R is called respectively WN, SN, or UN if so is any term in R.

Following [BK82, Klo92], we call a redex occurrence $u \subseteq t$ *perpetual* if $\infty(t) \Rightarrow \infty(s)$, where $t \xrightarrow{u} s$, and call u *critical* otherwise. Recall that a *perpetual strategy* is a function on terms which selects a perpetual redex in any ∞-term, and selects any redex (if any) otherwise [Bar84]. A redex (not an occurrence) is called *perpetual* iff its occurrence in any (admissible) context is perpetual.

Finally, let us recall the concept of *external* redexes [HL91]. These are redexes whose residuals or descendants can never occur in an argument of another redex. Any external redex is outermost, but not vice versa. (For example, consider the OTRS $R = \{f(x, g(y)) \to y, a \to g(b)\}$; then the first a in $f(a, a)$ is outermost but not external; the second a is external.) It is shown in [HL91] that any term not in normal form, in an OTRS, has an external redex; the same holds true for orthogonal ERSs [Kha94c], and similarly, for OCERSs as well.

Theorem 3.1 Any term not in normal form has an external redex, in an OCERS.

Definition 3.2 Let $P : t \twoheadrightarrow$ and $s \subseteq t$. We call P *internal* to s if it contracts redexes only in (the descendants of) s.

Definition 3.3 (1) Let $\infty(t)$, in an OCERS, and let $s \subseteq t$ be a smallest subterm of t such that $\infty(s)$ (i.e., $SN(e)$ for every proper subterm $e \subsetneq s$). We call s a *minimal perpetual subterm* of t, and call any external redex of s a *minimal perpetual redex* of t.

(2) Let F^∞_m be a one-step strategy which contracts a minimal perpetual redex in t if $\infty(t)$, and contracts any redex otherwise. Then we call F^∞_m a *minimal perpetual*

strategy. We call F_m^∞ *constricting* if for any F_m^∞-reduction $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots$ (i.e., constructed using F_m^∞), and for any i, $P_i^* : t_i \xrightarrow{u_i} t_{i+1} \xrightarrow{u_{i+1}} \ldots$ is internal to s_i, where $s_i \subseteq t_i$ is the minimal perpetual subterm containing u_i.

Note that F_m^∞ is not in general a computable strategy, as SN is undecidable already in orthogonal TRSs [Klo92].

Lemma 3.4 Let $\infty(t)$, let $s \subseteq t$ be a minimal perpetual subterm of t, and let $P : t \twoheadrightarrow \infty$ be internal to s. Then exactly one residual of any external redex u of s is contracted in P.

Proof. Let $t = C[s]$ and $s = C'[s_1, \ldots, u, \ldots, s_n]$, where C' consists of the symbols on the path from the top of s to u. If on the contrary P does not contract a residual of u, then every step of P takes place either in one of the s_i, or in the arguments of u (since u is external in s). Hence at least one of these subterms has an infinite reduction – a contradiction, since s is a minimal perpetual subterm. Since u is external, P cannot duplicate its residuals, hence P contracts exactly one residual of u.

Theorem 3.5 A minimal perpetual strategy F_m^∞ is a perpetual strategy, in an OCERS.

Proof. Suppose $\infty(t_0)$, let s_0 be a minimal perpetual subterm of t_0, and let $u \subseteq s_0$ be a minimal perpetuality redex. Let $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} t_2 \twoheadrightarrow \infty$ be internal to s_0. By Lemma 3.4, exactly one residual of u, say u_i, is contracted in P. Let $P_{i+1} : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots \xrightarrow{u_i} t_{i+1}$ and $P_{i+1}^* : t_{i+1} \xrightarrow{u_{i+1}} t_{i+2} \twoheadrightarrow \infty$ (i.e., $P : t_0 \overset{P_i}{\twoheadrightarrow} t_i \xrightarrow{u_i} t_{i+1} \overset{P_{i+1}^*}{\twoheadrightarrow}$). Then, by Theorem 2.5, $P = P_{i+1} + P_{i+1}^* \approx_L u + (P_i/u) + P_{i+1}^*$, i.e., u is a perpetual redex. Hence F_m^∞ is perpetual.

Definition 3.6 We call F_m^∞ the *leftmost* minimal perpetual strategy if in each term it contracts the leftmost minimal perpetual redex.[3]

Lemma 3.7 The leftmost minimal perpetual strategy is constricting, in an OCERS.

Proof. Let $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} t_2 \twoheadrightarrow \infty$ be a leftmost minimal perpetual reduction, and let $s_i \subseteq t_i$ be the leftmost minimal perpetual subterm of t_i. Since by Theorem 3.5 u_i is perpetual for the term s_i, the descendant of s_i is an ∞-term, hence contains s_{i+1}; and it is immediate that P is constricting.

Although we do not use it in the following, it is interesting to note that the constricting perpetual reductions are minimal w.r.t. Lévy's embedding relation $\trianglelefteq$, hence the name *minimal.*

The relations $\trianglelefteq, \approx_L$ and / are extended to co-initial possibly infinite reductions N, N' as follows. $N \trianglelefteq N'$, or equivalently, $N/N' = \emptyset$ if, for any redex v contracted

[3] Sørensen's and Melliès' strategies correspond to our leftmost and constricting minimal perpetual strategies, respectively. Gramlich's and Plaisted's strategies are defined for non-orthogonal rewrite systems, and they do not specify the perpetual redexes as external redexes of a minimal perpetual subterm.

in N, say $N = N_1 + v + N_2$, $v/(N'/N_1) = \emptyset$; and $N \approx_L N'$ iff $N \trianglelefteq N'$ and $N' \trianglelefteq N$. Here, for any infinite P, $u/P = \emptyset$ if $u/P' = \emptyset$ for some finite initial part P' of P, and P/Q is only defined for finite Q, as the reduction whose initial parts are residuals of initial parts of P under Q.

Theorem 3.8 Let $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} t_2 \twoheadrightarrow \infty$ be a constricting minimal perpetual reduction and let $Q : t_0 \twoheadrightarrow \infty$ be any infinite reduction such that $Q \trianglelefteq P$. Then $Q \approx_L P$.

Proof. Since P is constricting, there is a minimal perpetual subterm $s_0 \subseteq t_0$ such that P is internal to s_0. Since $Q \trianglelefteq P$, Q is internal to s_0 as well. By the construction, u_0 is an external redex in s_0, and by Lemma 3.4 exactly one residual u' of u_0 is contracted in Q. So let $Q : t_0 \overset{Q_j}{\twoheadrightarrow} t'_j \xrightarrow{u'} t'_{j+1} \overset{Q^*_{j+1}}{\twoheadrightarrow} \infty$. Then $Q \approx_L u_0 + Q_j/u_0 + Q^*_{j+1}$, and obviously $u_0 \trianglelefteq Q$. Similarly, since P is constricting, for any finite initial part P' of P, $P' \trianglelefteq Q$, and therefore $P \trianglelefteq Q$. Thus $Q \approx_L P$.

4 Two Characterizations of Critical Redexes

In this section, we give a very intuitive characterization of critical redex occurrences for OCERSs, generalizing Klop's characterization of critical redex occurrences for OTRSs [Klo92], and derive from it a characterization of perpetual redexes similar to Bergstra and Klop's perpetuality criterion for β-redexes [BK82]. Our proofs are surprisingly simple, yet the results are rather general and useful in applications. We need three simple lemmas first.

Lemma 4.1 Let $t \xrightarrow{u} s$, let $o \subseteq t$ be either in an argument of u or not overlapping with u, and let $o' \subseteq s$ be a u-descendant of o. Then $o' = o\theta$ for some substitution θ. If moreover o is a redex, then so is o' and $o \sim o'$.

Proof. Since u can be decomposed as a TRS-step followed by a number of substitution steps, it is enough to consider the cases when u is a TRS step and when it is an S-reduction step. If u is a TRS-step, or is an S-reduction step and o is not in its last argument, then o and o' coincide, hence $o \sim o'$ when o is a redex. Otherwise, $o' = o\theta$ for some substitution θ, and if o is a redex, we have again $o \sim o'$ by orthogonality and Lemma 2.7 since free variables of the substituted subterms cannot be bound in $o\theta$ (by the variable convention).

Lemma 4.2 Let s be a minimal perpetual subterm of t, and let $P : t \twoheadrightarrow \infty$ be internal to s. Then P has the form $P = t \twoheadrightarrow o \xrightarrow{u} e \twoheadrightarrow \infty$, where u is the descendant of s in o (i.e., a descendant of s necessarily becomes a redex and is contracted in P).

Proof. If on the contrary P does not contract descendants of s, then infinitely many steps of P are contracted in at least one of the proper subterms of s, contradicting its minimality.

Lemma 4.3 Let $P = u + P'$ be a constricting minimal perpetual reduction starting from t, in an OCERS, and let u be in an argument o of a redex $v \subseteq t$. Then P is internal to o.

Proof. Let $s \subseteq t$ be the minimal perpetual subterm containing u. By definition of minimal perpetual reductions, u is an external redex of s, hence s does not contain v. Since P is constricting, it is internal to s, and we have by orthogonality and Lemma 4.2 that s cannot overlap with the pattern of v. The lemma follows.

Definition 4.4 (1) Let $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \dots \xrightarrow{u_{k-1}} t_k$, and let $s_0, s_1, \dots, s_k$ be a chain of descendants of s_0 in along P (i.e, s_{i+1} is a u_i-descendant of $s_i \subseteq t_i$). Then, following [BK82], we call P ***passive*** w.r.t. $s_0, s_1, \dots, s_k$ if the pattern of u_i does not overlap with s_i (s_i may be in an argument of u_i or be disjoint from u_i) for $0 \leq i < k$. In the latter case, we call s_k a ***passive*** descendant of s_0. By Lemma 4.1, $s_k = s\theta$ for some substitution θ. We call θ a ***passive substitution*** or the ***P-substitution*** (w.r.t. $s_0, s_1, \dots, s_k$).

(2) Let t be a term in an OCERS, and let $s \subseteq t$. We call s a ***potentially infinite*** subterm of t if s has a passive descendant s' (along some reduction starting from t) s.t. $\infty(s')$. (Thus $\infty(s\theta)$ for some passive substitution θ.)

Theorem 4.5 Let $\infty(t)$ and let $t \xrightarrow{v} s$ be a critical step, in an OCERS. Then v erases a potentially infinite argument o (thus $\infty(o\theta)$ for some passive substitution θ).

Proof. Let $P : t = t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} t_2 \twoheadrightarrow \infty$ be a constricting minimal perpetual reduction, which exists by Theorem 3.5 and Lemma 3.7. Since v is critical, $SN(s)$, hence in particular P/v is finite. Let j be the minimal number such that $u_j/V_j = \emptyset$ and $u_j \notin V_j$, where $V_j = v/P_j$ and $P_j : t \twoheadrightarrow t_j$ is the initial part of P with j steps. (Below, V_j will denote both the corresponding set of residuals of v and its complete development.) By the Finite Developments theorem [KO95], no tail of P can contract only residuals of v; and since P/v is finite, such a j exists.

$$\begin{tikzcd}
t = t_0 \arrow[r, twoheadrightarrow] \arrow[d, "v"'] & t_l \arrow[r, twoheadrightarrow] \arrow[d, twoheadrightarrow, "V_l"'] & t_j \arrow[r, "u_j"] \arrow[d, twoheadrightarrow, "V_j"'] & t_{j+1} \arrow[r, twoheadrightarrow] \arrow[d, twoheadrightarrow] & P \\
s = s_0 \arrow[r, twoheadrightarrow] & s_l \arrow[r, twoheadrightarrow] & s_j \arrow[r, twoheadrightarrow, "\emptyset"'] & s_{j+1} \arrow[r, twoheadrightarrow, "\emptyset"'] & P/v
\end{tikzcd}$$

Since $u_j/V_j = \emptyset$ and $u_j \notin V_j$, there is a redex $v' \in V_j$ whose residual is contracted in V_j and erases (the residuals of) u_j. Since V_j consists of (possibly nested) residuals of a single redex $v \subseteq t_0$, the quantifiers in the pattern of v' cannot bind variables inside arguments of other redexes in V_j. Therefore v' is similar to its residual contracted in V_j by Lemma 2.7, and hence $u_j/v' = \emptyset$, implying that v' erases its argument o', say m-th from the left, containing u_j. By Lemma 4.3, the tail $P_j^* : t_j \twoheadrightarrow \infty$ of P is internal to o'.

Let $v_i \subseteq t_i$ be the predecessors of v' along P_j (so $v_0 = v$ and $v_j = v'$; note that a redex can have at most one predecessor), and let o_i be the m-th argument of v_i (thus $o' = o_j$). Note that $u_i \neq v_i$ since v_i has residuals. Let l be the minimal number such that u_l is in an argument of v_l (such an l exists as u_j is in an argument of v_j). Then all the remaining steps of P are in the same argument of v_l by Lemma 4.3, and it must be the m-th argument o_l of v_l (thus $\infty(o_l)$); but v' erases its m-th argument, implying by Lemma 2.7 that v_l also erases its m-th argument o_l. Further, by the choice of l, no steps of P are contracted inside v_i for $0 \leq i < l$, thus v_l is a passive descendent of v, and o_l is a passive descendant of o_0. Hence, by Lemma 4.1 $v \sim v_l$. Thus v erases a potentially infinite argument o_0 (since $\infty(o_l)$), and we are done.

Note in the above theorem that if the OCERS is an OTRSs, a potentially infinite argument is actually an ∞-term (since passive descendants are all identical), implying Klop's perpetuality lemma [Klo92]. O'Donnell's [O'Do77] lemma, stating that any term from which an innermost reduction is normalizing is strongly normalizable, is an immediate consequence of Klop's Lemma.

Corollary 4.6 Any redex whose erased arguments are closed SN-terms is perpetual, in OCERSs.

Proof. Immediate, as closed SN-terms cannot be potentially infinite subterms.

Note that Theorem 4.5 implies a general (although not computable) perpetual strategy: simply reduce a redex which does not erase a potentially infinite subterm. It is easy to check that the (maximal) perpetual strategies of Barendregt et al [BBKV76, Bar84] and de Vrijer [dVr87], and in general, the *limit* perpetual strategy of Khasidashvili [Kha94b, Kha94c], are special cases, as these strategies contract redexes whose arguments are in normal form, and no (sub)terms can be substituted in the descendants of these arguments. The minimal perpetual strategy, and hence the perpetual strategies of [Sør95, Mel96], are also special cases of the above general perpetual strategy.

We conclude this section with a characterization of perpetuality of erasing redexes, similar to the perpetuality criterion of β_K-redexes in [BK82].

Below, a substitution θ will be called SN iff $SN(x\theta)$ for every variable x.

Definition 4.7 We call a redex u *safe* (respectively, *SN-safe*) if either it is non-erasing, or else it is erasing and for any (resp. SN-) substitution θ, if $u\theta$ erases an ∞-argument, then the contractum of $u\theta$ is an ∞-term. (Note that, by Lemma 2.7, u is erasing iff $u\theta$ is, for any θ.)

Lemma 4.8 Let $\infty(t)$ and $s = t\theta$, in an OCERS. Then $\infty(s)$.

Proof. We prove that any infinite reduction $P : t\xrightarrow{u_0}t_1\xrightarrow{u_1}t_2 \twoheadrightarrow \infty$ can be simulated by some $Q : s = t\theta\xrightarrow{w_0}t_1\theta\xrightarrow{w_1}t_2\theta \twoheadrightarrow \infty$. it is enough to show that $t\xrightarrow{u}o$ implies $s = t\theta\xrightarrow{w}o\theta$ for some $w \subseteq s$, and to consider only the cases when u is a TRS-step or an S-reduction step. The first case is immediate, and the second follows from the Church-Rosser property for S-reductions.

Theorem 4.9 Any safe redex v, in an OCERS R, is perpetual.

Proof. Assume on the contrary that there is a context $C[]$ such that $t = C[v] \rightarrow s$ is a critical step. Let l be the minimal number such that, for some constricting minimal perpetual reduction $P : t = t_0\xrightarrow{u_0}t_1\xrightarrow{u_1}t_2 \twoheadrightarrow \infty$, the tail $P_l^* : t_l \twoheadrightarrow \infty$ of P is in an erased argument of a residual of v. Such an l exists by the proof of Theorem 4.5 (in the notation of that theorem, P_l^* is in an erased argument of $v_l \subseteq t_l$). Let v_l be the outermost among redexes in t_l which contain u_l (and therefore, whole P_l^*) in an erased argument, o_l, say m-th from the left (thus $\infty(o_l)$). By the proof of

Theorem 4.5, the m-th argument o of v is v-erased, $o_l = o\theta$, and $v_l = v\theta$ for some passive substitution θ.

We want to prove that the safety of v implies $\infty(s_l)$, hence $\infty(s)$, contradicting the assumption that $t \overset{v}{\longrightarrow} s$ is critical (see the diagram for Theorem 4.5). By the Finite Developments theorem, we can assume that s_l is obtained from t_l by contracting (some of) the redexes in V_l in the following order: (a) contract redexes in V_l disjoint from v_l; (b) contract redexes in V_l that are in the main arguments of v_l; (c) contract the residual v_l^* of v_l; (d) contract the remaining redexes, i.e., those containing v_l in a main (by the choice of v_l) argument. Since the parts (a) and (b) do not effect o_l, v_l^* erases an ∞-argument. (Recall from the proof of Theorem 4.5 that redexes in V_l are similar to their residuals contracted in any development of V_l.) Since $v_l = v\theta$ and redexes in (b) are in the substitution part of v_l, $v_l^* = v\theta^*$ for some substitution θ^*, hence its contractum e is infinite by safety of v. By the choice of v_l, e has a descendant e' in s_l after the part (d), and by Lemma 4.1, $e' = e\theta'$ for some substitution θ'. By Lemma 4.8, $\infty(e)$ implies $\infty(e')$, hence $\infty(s_l)$ – a contradiction.

5 Applications

We now give a number of applications of our perpetuality criteria, demonstrating their usefulness and powerfulness. Below, in some of the examples, we will use the conventional λ-calculus notation [Bar84]; and by the *argument* of a β-redex $(\lambda x.s)o$ we will mean its second argument o.

5.1 The restricted orthogonal λ-calculi

Let us call *orthogonal restricted λ-calculi* (ORLC) the calculi that are obtained from the λ-calculus by restricting the β-rule (by some conditions on arguments and contexts) and that are orthogonal CERSs. Examples include the λ_I-calculus, the call-by-value λ-calculus [Plo75], as well as a large class of typed λ-calculi.

If R is an ORLC, then in the proofs of Theorem 4.5 and Theorem 4.9, the P_l-substitution (and in general, any passive substitution along a constricting perpetual reduction) is SN. This can be proved similarly to [BK82] (see Proposition 2.8), since, in the terminology of [BK82] and the notation of Theorem 4.5 and Theorem 4.9:

- P_l is SN-substituting (meaning that the arguments of contracted β-redexes are SN). This is immediate from the minimality of P_l, and
- P_l is *simple* (meaning that no subterms can be substituted in the substituted, during the previous steps, subterms). This follows immediately from externality, w.r.t. the chosen minimal perpetual subterm, of minimal perpetual redexes (P_l is standard).

Hence, we have the following two corollaries, of which the latter is a mere extension of Bergstra-Klop criterion [BK82] (in the case of β-redexes, the converse statement is much easier to prove, see [BK82]).

Corollary 5.1 Let $\infty(t)$ and let $t \overset{v}{\longrightarrow} s$ be a critical step, in an ORLC. Then v erases a potentially infinite argument o such that $\infty(o\theta)$ for some passive SN-substitution θ.

Corollary 5.2 Any SN-safe redex v, in an ORLC, is perpetual.

Note that these corollaries are not valid for OCERSs in general since, unlike ORLC, passive substitutions along constricting perpetual reductions need not be SN in OCERSs: Let $R = S \cup \{\sigma xAB \to Sx\omega(A/x)B,\ E(x) \to a\}$ where $\omega = \lambda x.Ap(x,x)$. Then the step $\sigma xAp(x,x)E(x) \to \sigma xAp(x,x)a$ is SN-safe (as it only erases a variable), but is critical as can be seen from the following diagram, of which the bottom part is the only reduction starting from $\sigma xAp(x,x)a$:

$$\begin{tikzcd}
\sigma xAp(x,x)E(x) \arrow[r, "\sigma"'] \arrow[d, "E"'] & SxwE(Ap(x,x)) \arrow[r, "S"'] \arrow[d, "E"'] & E(Ap(w,w)) \arrow[r, "\beta"'] \arrow[d, "E"'] & E(Ap(w,w)) \arrow[r, "\beta"'] \arrow[d, "E"'] & {} \\
\sigma xAp(x,x)a \arrow[r, "\sigma"'] & Sxwa \arrow[r, "S"'] & a \arrow[r, two heads, "\emptyset"'] & a \arrow[r, two heads, "\emptyset"'] & {}
\end{tikzcd}$$

5.2 Plotkin's call-by-value λ-calculus

Plotkin [Plo75] introduced the *call-by-value* λ-calculus, λ_V, which restricts the usual λ-calculus by allowing the contraction of redexes whose arguments are *values*, i.e., either abstractions $\lambda x.t$ or variables (we assume that there are no δ-rules in the calculus). Let the *lazy* call-by-value λ-calculus λ_{LV} be obtained from λ_V by allowing only call-by-value redexes that are not in the scope of a λ-occurrence (λ_{LV} is enough for computing values in λ_V, see Corollary 1 in [Plo75]). Then it follows from Corollary 5.1, as well as from Corollary 5.2, that any λ_{LV}-redex is perpetual, hence λ_{LV} is UN. Indeed, let $v = (\lambda x.s)o$ be a λ_{LV}-redex. Then, if o is a variable, then it is immediate that v cannot be critical, and if o is an abstraction, any of its instances is an abstraction too, hence is a λ_{LV}-normal form. This is not surprising, however, as λ_{LV}-redexes are disjoint,[4] and there is no duplication or erasure of (admissible) redexes.

5.3 De Groote's β_{IS}-reduction

De Groote [dGr93] introduced β_S-reduction on λ-terms by the following rule: $\beta_S : (((\lambda x.M)N)O) \to ((\lambda x.(MO)N)$, where $x \notin FV(M,O)$. He proved that the β_{IS}-calculus is uniformly normalizing. Clearly, this is an immediate corollary of Theorem 4.5 as the β_S- and β_I-rules are non-erasing (note that these rules do not conflict because of the conditions on bound variables). Using this result, the author proves strong normalization of a number of typed λ-calculi.

5.4 Böhm &Intrigila's λ-δ_k-calculus

Böhm and Intrigila [BI94] introduced the λ-δ_k-calculus in order to study UN solutions to fixed point equations, in the $\lambda\eta$-calculus. Since the K-redexes are the source of failure of the UN property in the $\lambda(\eta)$-calculus, they define a 'restricted K

[4] if u, v are redexes in a term t and $u = (\lambda x.e)o$, then $v \notin e$ because of the main λ of u, and $v \notin o$ since o is either a variable or an abstraction; orthogonality of λ_{LV} follows from a similar argument.

combinator' δ_K by the following rule: $\delta_K AB \to A$, where B can be instantiated to closed λ-δ_k-normal forms (possibly containing δ_K constants; such a reduction is still well defined). λ-δ_k-terms are λ_I-terms with the constant δ_K. The authors show that the λ-δ_k-calculus has the UN property. This result follows from Corollary 4.6 only if the η-rule is dropped. However, Klop shows in [Klo80] that η-redexes are perpetual, and we hope that our results can be generalized to weakly-orthogonal CERSs (and thus cover the η-rule since η-redexes are non-erasing) using van Oostrom and van Raamsdonk's technique for simulating $\beta\eta$ reductions with β-reductions [OR94].

5.5 Honsell &Lenisa's β_{N°-calculus

Honsell and Lenisa [HL93] define a similar reduction, β_{N°-reduction, on λ-terms by the following rule: $\beta_{N^\circ} : \ (\lambda x.A)B \to (B/x)A$, where B can be instantiated to a closed β-normal form. We have immediately from Corollary 4.6 that β_{N° is UN. Note however that the later does not follow (at least, without an extra argument) from Bergstra and Klop's characterization of perpetual β_K-redexes [BK82] as $\beta_{N^\circ} \subset \beta$ but not conversely. (If t has an infinite β_{N°-reduction and $t \overset{u}{\to} s$ is a β_{N°-step, then the Bergstra-Klop criterion implies existence of an infinite *β-reduction* starting from s, not existence of an infinite β_{N°-reduction.)

6 Concluding Remarks

We have obtained two criteria for perpetuality of redexes in orthogonal CERSs, and demonstrated their usefulness in applications. We claim that our results are also valid for Klop's orthogonal *substructure* CRSs [KOR93].

However, they cannot be generalized (at least, directly) to orthogonal *Pattern Rewrite Systems* (OPRSs) [Nip93], as witnessed by the following example due to van Oostrom [Oos97]. It shows that already the Conservation Theorem fails for OPRSs (i.e., non-erasing steps need not be perpetual): Let $R = \{g(M.N.X(x.M(x), N)) \to_g X(x.I, \Omega),\ @(\lambda(x.M(x)), N) \to_\beta M(N)\}$ where $\Omega = @(\lambda(x.xx), \lambda(x.xx))$. Then $g(M.N.@(\lambda(x.M(x)), N)) \to_\beta g(M.N.M(N))$ is non-erasing but critical, as can be seen from the following diagram, of which the bottom part is the only reduction starting from $g(M.N.M(N))$. Such strange behavior arises due to the λ_K-reduction steps in the substitution calculus, which are invisible in a PRS reduction step.

$$\begin{tikzcd}
g(M.N.@(\lambda(x.M(x)), N)) \arrow[r, "g"'] \arrow[d, "\beta"'] & @(\lambda(x.I), \Omega) \arrow[r, "\Omega", "\beta"'] \arrow[d, "\beta"'] & @(\lambda(x.I), \Omega) \arrow[r, "\Omega", "\beta"'] \arrow[d, "\beta"'] & \cdots \\
g(M.N.M(N)) \arrow[r, "g"'] & I \arrow[r, two heads, "\emptyset"'] & I \arrow[r, two heads, "\emptyset"'] & \cdots
\end{tikzcd}$$

Acknowledgments We thank Vincent van Oostrom for useful comments, and for the counterexample.

References

[Acz78] Aczel P. A general Church-Rosser theorem. Preprint, University of Manchester, 1978.

[Bar84] Barendregt H. P. The Lambda Calculus, its Syntax and Semantics. North-Holland, 1984.

[Bar92] Barendregt H. P. Lambda calculi with types. In: Handbook of Logic in Computer Science, vol. 2, S. Abramsky, D. Gabbay, and T. Maibaum eds. Oxford University Press, 1992, p. 117-309.

[BBKV76] 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, 1976, p. 13-53.

[BK82] Bergstra J. A., Klop J. W. Strong normalization and perpetual reductions in the Lambda Calculus. J. of Information Processing and Cybernetics, 18:403-417, 1982.

[BI94] Böhm C., Intrigila B. The Ant-Lion paradigm for strong normalization. Information and Computation, 114(1):30-49, 1994.

[CR36] Church A., Rosser, J. B. Some Properties of Conversion. Transactions of the American Mathematical Society, 39:472-482, 1936.

[GK96] Glauert J. R. W., Khasidashvili Z. Minimal relative normalization in orthogonal expression reduction systems. In Proc. of the 16^{th} International Conference on Foundations of Software Technology and Theoretical Computer Science, FST&TCS'96, Springer LNCS, vol. 1180, V. Chandru, ed. 1996, p. 238-249.

[Gra96] Gramlich B. Termination and confluence properties of structured Rewrite Systems. Ph.D. Thesis, Kaiserslautern University, 1996.

[dGr93] De Groote P. The conservation theorem revisited. In: Proc. of the 1^{st} International Conference on Typed Lambda Calculi and Applications. Springer LNCS, vol. 664, M. Bezem, J. F. Groote, eds. Utrecht, 1993, p. 163-178.

[HL93] Honsell F., Lenisa M. Some results on the full abstraction problem for restricted lambda calculi. In Proc. of the 18^{th} International Symposium on Mathematical Foundations of Computer Science, MFCS'93, Springer LNCS, vol. 711, A. M. Borzyszkowski, S. Sokolowski, eds. 1993, p. 84-104.

[HL91] Huet G., Lévy J.-J. Computations in Orthogonal Rewriting Systems. In: Computational Logic, Essays in Honor of Alan Robinson, J.-L. Lassez and G. Plotkin, eds. MIT Press, 1991, p. 394-443.

[Kha92] Khasidashvili Z. The Church-Rosser theorem in Orthogonal Combinatory Reduction Systems. Report 1825, INRIA Rocquencourt, 1992.

[Kha94] Khasidashvili Z. On higher order recursive program schemes. In: Proc. of the 19^{th} International Colloquium on Trees in Algebra and Programming, CAAP'94, Springer LNCS, vol. 787, S. Tison, ed. Edinburgh, 1994, p. 172-186.

[Kha94b] Khasidashvili Z. Perpetuality and strong normalization in orthogonal term rewriting systems. In: Proc. of 11^{th} Symposium on Theoretical Aspects of Computer Science, STACS'94, Springer LNCS, vol. 775, P. Enjalbert, E. W. Mayr, and K. W. Wagner, eds. Caen, 1994, p. 163-174.

[Kha94c] Khasidashvili Z. The longest perpetual reductions in orthogonal expression reduction systems. In: Proc. of the 3^{rd} International Conference on Logical Foundations of Computer Science, LFCS'94, Springer LNCS, vol. 813, Nerode A., Matiyasevich Yu. V. eds. 1994. p. 191-203.

[KO95] Khasidashvili Z, van Oostrom V. Context-sensitive conditional expression reduction systems. In proc. of the International Workshop on Graph Rewriting and Computation, SEGRAGRA'95. In Electronic Notes in Computer Science, A. Corradini, U. Montanari, eds. Elsevier Science B.V., August, 1995, p. 141-150.

[Klo80] Klop J. W. Combinatory Reduction Systems. Mathematical Centre Tracts n. 127, CWI, Amsterdam, 1980.

[Klo92] Klop J. W. Term Rewriting Systems. In: Handbook of Logic in Computer Science, vol. 2, S. Abramsky, D. Gabbay, and T. Maibaum eds. Oxford University Press, 1992, p. 1-116.

[KOR93] Klop J. W., van Oostrom V., van Raamsdonk F. Combinatory reduction systems: introduction and survey. In: To Corrado Böhm, Special issue of Theoretical Computer Science, 121:279-308, 1993.

[Lév80] Lévy J.-J. Optimal reductions in the Lambda-calculus. In: To H. B. Curry: Essays on Combinatory Logic, Lambda-calculus and Formalism, Hindley J. R., Seldin J. P. eds, Academic Press, 1980, p. 159-192.

[Mel96] Melliès P.-A. Description Abstraite des Systèmes de Réécriture. Thèse de l'Université Paris 7, 1996.

[Ned73] Nederpelt R. P. Strong Normalization for a typed lambda-calculus with lambda structured types. Ph.D. Thesis, Eindhoven, 1973.

[Nip93] Nipkow T. Orthogonal higher-order rewrite systems are confluent. In: Proc. of the 1^{st} International Conference on Typed Lambda Calculus and Applications, TLCA'93, Springer LNCS, vol. 664, Bezem M., Groote J.F., eds. Utrecht, 1993, p. 306-317.

[O'Do77] O'Donnell M. J. Computing in systems described by equations. Springer LNCS, vol. 58, 1977.

[Oos94] Van Oostrom V. Confluence for Abstract and Higher-Order Rewriting. Ph.D. Thesis, Free University, Amsterdam, 1994.

[Oos97] Van Oostrom V. Personal communication, 1997.

[OR94] Van Oostrom V., van Raamsdonk F. Weak orthogonality implies confluence: the higher-order case. In: Proc. of the 3^{rd} International Conference on Logical Foundations of Computer Science, LFCS'94, Springer LNCS, vol. 813, Nerode A., Matiyasevich Yu. V. eds. 1994. p. 379-392.

[Pkh77] Pkhakadze Sh. Some problems of the Notation Theory (in Russian). Proceedings of I. Vekua Institute of Applied Mathematics of Tbilisi State University, Tbilisi, 1977.

[Pla93] Plaisted D. A. Polynomial time termination and constraint satisfaction tests. In: Proc. of the 5^{th} International Conference on Rewriting Techniques and Applications, RTA'93, Springer LNCS, vol. 690, C. Kirchner, ed. 1993, p. 405-420.

[Plo75] Plotkin G. Call-by-name, call-by-value and the λ-calculus. Theoretical Computer Science, 1:125-159, 1975.

[Raa96] Van Raamsdonk F. Confluence and normalisation for higher-order rewriting. Ph.D. Thesis, Free University, Amsterdam, 1996.

[Sør95] Sørensen M. H. Properties of infinite reduction paths in untyped λ-calculus. In: Proc. of Tbilisi Symposium on Logic, Language and Computation, TSLLC'95. Selected papers. J. Ginzburg, Z. Khasidashvili, J.J. Lévy, C. Vogel, E. Vallduví, eds. SiLLI publications, CSLI, Stanford, 1997. To appear.

[dVr87] De Vrijer R. C. Surjective pairing and strong normalization: two themes in lambda calculus. Ph.D. Thesis, Amsterdam, 1987.

[Wol93] Wolfram D. The causal theory of types. Cambridge Tracts in Theoretical Computer Science, vol. 21, Cambridge University Press, 1993.

This article was processed using the LaTeX macro package with LLNCS style