Discrete Normalization and Standardization in Deterministic Residual Structures

Zurab Khasidashvili and John Glauert

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

Abstract. We prove a version of the *Standardization Theorem* and the *Discrete Normalization Theorem* in *stable Deterministic Residual Structures*, Abstract Reduction Systems with axiomatized notions of *residual*, which model orthogonal rewrite systems. The latter theorem gives a strategy for construction of reductions Lévy-equivalent (or permutation-equivalent) to a given, finite or infinite, *regular* (or *semi-linear*) reduction, based on the *neededness* concept of Huet and Lévy. This and other results of this paper add to the understanding of Lévy-equivalence of reductions, and consequently, Lévy's reduction space. We demonstrate how elements of this space can be used to give denotational semantics to known functional languages in an abstract manner.

1 Introduction

Long ago, Curry and Feys [CuFe58] proved that repeated contraction of leftmost-outermost redexes in any normalizable λ-term eventually yields its normal form, even if the term possesses infinite reductions as well. The reason is that such redexes are *needed* in every normalizable term t, i.e., they are contracted in every normalizing reduction starting from t. Based on this observation, Huet and Lévy defined a general normalizing strategy, the *needed strategy*, for Orthogonal Term Rewriting Systems (OTRSs). They showed that any term t not in normal form has a needed redex, and that repeated contraction of needed redexes in t leads to its normal form whenever there is one. This seminal work has been extended in many directions.

Barendregt et al. [BKKS87] studied neededness w.r.t. normal forms as well as w.r.t. head-normal forms, in the λ-calculus, proving correctness of the two needed strategies for computing normal forms and head-normal forms, respectively. Kennaway and Sleep [KeSl89] generalized the needed strategy to orthogonal Combinatory Reduction systems (CRSs) of Klop [Klo80]. Maranget [Mar92] studied a strategy that computes a weak head-normal form of a term in an OTRS. Normalization w.r.t. another interesting set of 'normal forms', that of constructor head-normal forms in constructor OTRSs, is studied by Nöcker [Nök94].

Khasidashvili defined a similar normalizing strategy, the *essential* strategy, for the λ-calculus [Kha88], OTRSs [Kha93], and Orthogonal Expression Reduction Systems (OERS) [Kha94], a form of higher-order rewriting similar to Klop's CRSs

* This work was supported by the Engineering and Physical Sciences Research Council of Great Britain under grant GR/H 41300

(which subsumes Term Rewriting and the λ-calculus) [Kha92]. Sekar and Ramakrishnan [SeRa93] study a normalizing strategy which in each multi-step contracts a *necessary* set of redexes. A different approach to normalization is developed in Kennaway [Ken89] and in Antoy and Middeldorp [AnMi94]. Antoy et al. [AEH94] designed a needed narrowing strategy. Gardner [Gar94] described a *complete* way of encoding neededness information using a type assignment system, in the λ-calculus. Kennaway et al. [KKSV95] studied needed strategies for infinitary OTRSs.

In [GlKh94], the present authors found natural conditions on a set $\mathcal{S}$ of terms, in an OERS, called *stability*, that are necessary and sufficient for the following *Relative Normalization* (RN for short) theorem to hold: each $\mathcal{S}$-normalizable term not in $\mathcal{S}$ (not in $\mathcal{S}$-*normal form*) has at least one $\mathcal{S}$-needed redex, and repeated contraction of $\mathcal{S}$-needed redexes in a term t will lead to an $\mathcal{S}$-normal form of t whenever there is one. Roughly, $\mathcal{S}$ is stable if it is closed under reduction (this condition can be relaxed slightly) and any step u entering $\mathcal{S}$ is $\mathcal{S}$-needed.

In [GlKh96], the authors further generalized the theory of relative normalization by abstracting from the structure of terms. They study relative normalization in *Deterministic Residual Structures* (DRSs), which are *Abstract Reduction Systems* (ARSs) where the residual relation on redexes is axiomatized in a way that enables one to define permutation-equivalence on reductions. They proved the RN theorem for *stable* DRS, for *regular* stable sets $\mathcal{S}$ of final terms. Here a stable set $\mathcal{S}$ is *regular* if $\mathcal{S}$-unneeded redexes cannot duplicate $\mathcal{S}$-needed ones.

Here we study reductions from a term t to an arbitrary term s, assuming such a reduction exists. If the set $\{s\}$ is stable, such as when s is a normal form, then existing theory can be applied. Otherwise, if the reduction graph G_s of s (which consists of terms to which s is reducible) is stable and *regular*, then the task can be reduced to construction of a *minimal*, or rather a *least* (w.r.t. Lévy's ordering $\trianglelefteq$ on reductions) G_s-normalizing reduction [GlKh94a], since it must end at s. However, as we already know [GlKh94], G_s need not be stable, mainly because of *syntactic accidents* [Lév80]. For example, the graph $\{I(x), x\}$ of $s = I(x)$, in the TRS with the rule $I(x) \to x$, is not stable, since $t = I(I(x))$ can be reduced to s by contracting either redex. Neither of the two redexes in t is G_s-needed, and neither of the two reductions $u : t \overset{I(I(x))}{\twoheadrightarrow} s$ and $v : t \overset{I(x)}{\twoheadrightarrow} s$ is G_s-least – we have $u \ntrianglelefteq v$, $v \ntrianglelefteq u$, and $u \not\approx_L v$.

A constructive approach is to develop a neededness theory relative to *particular* reductions $P : t \twoheadrightarrow s$ (constructed according to some strategy), or more precisely, relative to Lévy-equivalence classes $\langle P \rangle_L$ of reductions P, which are elements of Lévy's reduction space ordered by $\trianglelefteq$. (Hence the name – *discrete* normalization.) We cannot aim at developing a needed strategy constructing *a* reduction from t to s because a redex in t contracted in every reduction from t to s may not exist, as is the case for the terms $t = I(I(x))$ and $s = I(x)$.[2] Furthermore, the task of construction of a reduction Lévy-equivalent to a given one makes sense also for infinite reductions.

In this paper we indeed develop a neededness theory relative to particular *semi-linear*, or *regular*, reductions, which enables us to characterize reductions in $\langle P \rangle_L$ for any such reduction $P : t \twoheadrightarrow$, finite or infinite, in stable DRSs. (By an 'infinite' reduction we mean a reduction of length ω.) We obtain two main results: the

[2] However, 'normalization via necessary sets' approach of [SeRa93] may be useful here.

Standardization Theorem, and the *Discrete Normalization Theorem*. The first states that every finite semi-linear reduction P can, in some canonical way, be transformed into a *standard* or *self-essential* reduction $SE(P)$ Lévy-equivalent to P, where self-essentiality means that 'omission' of a step in $SE(P)$ would yield a reduction no longer Lévy-equivalent to P. We also show that for infinite semi-linear reductions P, standard reductions Lévy-equivalent to P need not exist. This concept of standardization is the best approximation to the outside-in left-to-right concept of standardization [Bar84, Klo80], since we do not have any nesting reduction imposed on redexes, unlike ARSs of [GLM92], and there is no concept of 'left' or 'right' occurrences in DRSs. However, our standard reductions can be used successfully as canonical representatives in their Lévy-equivalence classes, without being unique representatives. The Discrete Normalization (DN) theorem states that for any P, repeated simultaneous contraction of all P-erased redexes yields a reduction in $\langle P\rangle_L$, and if P is semi-linear and finite, then contraction of P*-needed* P*-erased* redexes eventually terminates by constructing a reduction in $\langle P\rangle_L$. Here a redex is P-needed if at least one residual of it is contracted by any reduction in $\langle P\rangle_L$, and is P-erased if it does not have residuals in some later term of P. P is semi-linear if its P-unneeded steps do not duplicate P-needed redexes. In orthogonal (acyclic) syntactic rewrite systems, all reductions are semi-linear.

Since every functional programming language is given by a (very often orthogonal) syntactic rewrite system R and a *deterministic* strategy F (e.g. the lazy, call-by-name, and several eager strategies in the λ-calculus), the *value* of a term (program) t is defined via its computation according to F. In semantics, values of terms are usually defined via some kind of finite or infinite trees, such as Böhm-trees [Bar84], constructed from symbols from the alphabet and a special constant $\perp$ to denote *undefined* (in some sense) terms. The strategies F compute these trees. So, our DN theorem characterizes computations of a term which converge to its value, where value can be defined in various ways. We will describe an abstract approach to semantics of orthogonal rewrite systems, showing how the conventional tree semantics of such systems can be recovered without any assumption of syntax, using Lévy's reduction space which we study here.

The paper is organized as follows. In the next section, we recall the definition of DRSs. In Section 3, we study relative properties of discrete neededness, essentiality and erasure. Section 4 is devoted to the standardization and related results, which are used in Section 5 to prove the Discrete Normalization Theorem, the main result of this paper. Section 6 is devoted to semantics of DRSs. Conclusions appear in Section 7. More details can be found in [KhGl96].

2 Deterministic Residual Structures

In this section we recall *Deterministic Residual Structures* (DRSs), introduced in [GlKh96] but largely based on [Lév80]. DRSs are *Abstract Reduction Systems* (ARSs) with axiomatized notions of residual.

A definition and a survey of results about ARSs can be found in [Klo92]. Our definition is slightly different, and follows that of Hindley [Hin64]. An ARS is a triple $A = (Ter, Red, \rightarrow)$ where *Ter* is a set of *terms*, ranged over by t, s, o, e; *Red* is a set of *redexes* (or *redex occurrences*), ranged over by u, v, w; and $\rightarrow$: $Red \mapsto (Ter \times Ter)$ is a function such that for any $t \in Ter$ there is only a finite set of $u \in Red$ such that $\mapsto (u) = (t, s)$, written $t \xrightarrow{u} s$. This set will be known as the redexes of term t, where $u \in t$ denotes that u is a member of the redexes of t and $U \subseteq t$ denotes that U is a subset of the redexes. Note that $\rightarrow$ is a *total* function, so one can identify u with the triple $t \xrightarrow{u} s$. (Klop's ARSs are pairs $(Ter, \rightarrow)$, and do not allow one to distinguish transitions with the same source and target terms.) A *reduction* is a sequence $t \xrightarrow{u_1} t_2 \xrightarrow{u_2} \ldots$. Reductions are denoted by P, Q, N. We write $P : t \twoheadrightarrow s$ or $t \overset{P}{\twoheadrightarrow} s$ if P denotes a reduction (sequence) from t to s, and write $P : t \twoheadrightarrow$ if P denotes a (finite or infinite) reduction starting from t. $|P|$ denotes the length of P. $P + Q$ denotes the concatenation of P and Q. $P \leq Q$ denotes that P is an initial part of Q. We use U, V, W to denote sets of redexes of a term.

DRSs descend from Stark's *Determinate Concurrent Transition Systems* (DCTSs) [Sta89] and *Abstract Reduction Systems* of Gonthier et al. [GLM92]. Unlike DCTSs, the residual relation in DRSs may be duplicating, and unlike ARSs of [GLM92], we do not have a nesting relation on redexes. Several refined concepts of abstract rewriting are studied in [Oos94, Mel96, Raa96]. DRS are intended to model all orthogonal term and graph rewrite systems, both first and higher-order, such as [Lév80, HuLé91, Klo80, Kha92, Nip93, Lam90, Kat90, SPE93, Oos94, AsLa96, Raa96], and other conflict-free transition systems with a reasonable level of abstraction, while say ARSs of [GLM92] do not cover e.g., orthogonal term graph rewriting systems [SPE93].

Definition 2.1 ([GlKh96]) **(Deterministic Residual Structure)** A *Deterministic Residual Structure* (DRS) is a pair $R = (A, /)$, where A is an ARS and $/$ is a *residual* relation on redexes relating redexes in the source and target term of every reduction $t \xrightarrow{u} s \in A$, such that for $v \in t$, the set v/u of *residuals of v under u* is a set of redexes of s; a redex in s may be a residual of only one redex in t under u, and $u/u = \emptyset$. If v has more than one u-residual, then u *duplicates* v. If $v/u = \emptyset$, then u *erases* v. A redex of s which is not a residual of any $v \in t$ under u is said to be *created* by u. The set of residuals of a redex under any reduction is defined by transitivity.

A *development* of a set U of redexes in a term t is a reduction $P : t \twoheadrightarrow$ that only contracts residuals of redexes from U; the development P is *complete* if U/P, the set of residuals under P of redexes from U, is empty $\emptyset$. Development of $\emptyset$ is identified with the empty reduction. U will also denote a complete development of $U \subseteq t$. The residual relation satisfies the following two axioms, called *Finite Developments (FD)* [GLM92] and *acyclicity* (which appears as axiom (4) in [Sta89]):

- [FD] All developments are terminating; all co-initial complete developments of the same set of redexes end at the same term; and residuals of a redex under all complete co-initial developments of a set of redexes are the same.
- [acyclicity] Let $u, v \in t$, let $u \neq v$, and let $u/v = \emptyset$. Then $v/u \neq \emptyset$.

Note that [acyclicity] re-introduces some kind of 'nesting' relation on redexes, but we do not even require this relation to be transitive.

As in [Lév80, HuLé91, Sta89], the residual relation on redexes in a DRS R is extended to all co-initial finite reductions as follows: $(P_1 + P_2)/Q = P_1/Q + P_2/(Q/P_1)$ and $P/(Q_1 + Q_2) = (P/Q_1)/Q_2$, and *Lévy-* or *permutation-equivalence* is defined as the smallest relation on co-initial reductions satisfying: $U + V/U \approx_L V + U/V$ and $Q \approx_L Q' \Rightarrow P+Q+N \approx_L P+Q'+N$, where U and V are complete developments of redex sets in the same term. Further, one defines $P \trianglelefteq Q$ iff $P/Q = \emptyset$, and can show that $P \approx_L Q$ iff $P \trianglelefteq Q$ and $Q \trianglelefteq P$; and $P \trianglelefteq Q$ iff $Q \approx_L P + N$ for some N. Intuitively, $P \approx_L Q$ means that P can be obtained from Q by a number of permutations of adjacent steps, therefore 'Q and P do the same work'; and $P \trianglelefteq Q$ means that P does less work than Q, the difference being Q/P, so $P + Q/P \approx_L Q$. Unlike [Lév80, HuLé91], we do not consider complete developments of sets of residuals of a redex as multi-steps, so P/Q is only unique up to the choice of particular complete developments of such sets. (However, P/Q is defined more precisely than up to $\approx_L$.) The above relations can equivalently be defined also using Klop's method of commutative diagrams [Klo80, Bar84]. We will often use Klop's diagrams in proofs.

Lévy-equivalence extends to infinite reductions, denoted by $t \twoheadrightarrow \infty$, as follows. First, for any $u \in t \overset{P}{\twoheadrightarrow}$, define $u/P = \emptyset$ if $u/P' = \emptyset$ for a finite initial part P' of P. Now, for any $P, Q : t \twoheadrightarrow$, define $P/Q = \emptyset$, or equivalently $P \trianglelefteq Q$, if for any redex v contracted in P, say $P = P' + v + P''$, $v/(Q/P') = \emptyset$; and define $P \approx_L Q$ iff $P \trianglelefteq Q$ and $Q \trianglelefteq P$. P/Q is only defined for finite Q, as the reduction whose initial parts are residuals of initial parts of P under Q.

Here we are only interested in *stable* DRSs where a given redex cannot arise from two unrelated sources:

Definition 2.2 [GlKh96] A DRS is *stable*, SDRS, if the following axiom is satisfied:
• [stability] Let $t \overset{u}{\rightarrow} e$, $t \overset{v}{\rightarrow} s$, $u \neq v$, and let u create $w \in e$. Then the redexes in $w/(v/u)$ are not u/v-residuals of redexes of s, i.e., they are created by u/v.

The essence of stability is better understood via Lemma 2.2 below which expresses the preservation of *externality* during redex creation. For syntactic systems externality is a natural concept relating to overlap between components of terms involved in reduction steps. In an abstract setting it expresses the absence of shared (residuals of) redexes.

Definition 2.3 ([GlKh96]) • Let $u \in U \subseteq t$ and $P : t \twoheadrightarrow$. We call P *external* to U (resp. u) if P does not contract residuals of redexes in U (resp. residuals of u).

• Let $P : t_0 \overset{P_i}{\twoheadrightarrow} t_i \overset{u_i}{\rightarrow} t_{i+1} \twoheadrightarrow$ and $Q : t_0 = s_0 \overset{Q_j}{\twoheadrightarrow} s_j \overset{v_j}{\rightarrow} s_{j+1} \twoheadrightarrow$. We call P *external* to Q if for any i, j, $u_i/(Q_j/P_i) \cap v_j/(P_i/Q_j) = \emptyset$ (see the diagram, where $U_{i,j} = u_i/(Q_j/P_i)$ and $V_{i,j} = v_j/(P_i/Q_j)$).

$$
\begin{tikzcd}
t_0 \arrow[r, two heads, "P_i"] \arrow[d, two heads, "Q_j"'] & t_i \arrow[r, "u_i"] \arrow[d, two heads, "Q_j/P_i"'] & t_{i+1} \arrow[d, two heads] \\
s_j \arrow[r, two heads, "P_i/Q_j"] \arrow[d, "v_j"'] & \cdot \arrow[r, two heads, "U_{i,j}"] \arrow[d, two heads, "V_{i,j}"'] & \cdot \arrow[d, two heads] \\
s_{j+1} \arrow[r, two heads] & \cdot \arrow[r, two heads] & \cdot
\end{tikzcd}
$$

Obviously, P is external to the set U iff it is external to any development of U, and is external to a redex u iff it is external to the reduction contracting u. Note that a reduction external to one complete development of U need not be external to all developments of U, and in general, externality is not invariant under $\approx_L$. For, consider a TRS $R = \{a \to a', f(x) \to b, g(x) \to c\}$, a term $t = f(g(a))$, and reductions $P : t \xrightarrow{a} f(g(a')) \xrightarrow{f} b$, $Q : t \xrightarrow{a} f(g(a')) \xrightarrow{g} f(c)$, and $N : t \xrightarrow{g} f(c)$. Then we have $Q \approx_L N$, P is external to N, but not to Q; and P is not external to $U = \{a, g(a)\}$.

Lemma 2.1 ([GlKh96, GlKh96a]) Let $P : t \twoheadrightarrow s$ be external to a complete development N of a set of redexes $U \subseteq t$, in a DRS, and let $Q : t \twoheadrightarrow o$, then P/Q is external to N/Q.

Lemma 2.2 (Stability Lemma) ([GlKh96]) Let $P : t \twoheadrightarrow s$ be external to $Q : t \twoheadrightarrow e$, in a stable DRS, and let P create redexes $W \subseteq s$. Then the residuals $W/(Q/P)$ of redexes in W are created by P/Q, and Q/P is external to W.

As with the Stability Lemma, the following lemma extends the [acyclicity] property from one step external reductions to arbitrary finite external reductions.

Lemma 2.3 (Acyclicity Lemma) ([GlKh96a, KhGl96]) Let P, N be co-initial finite reductions in a DRS, and let P be external to N. Then $N \not\approx_L P$.

3 Existence of Discretely Needed Redexes

In this section, we characterise some properties of a redex with respect to possible reductions in a DRS. In particular, we define P-*needed* redexes, for any reduction P, and then prove existence of P-needed redexes for any finite P in a stable DRS.

3.1 Discrete Notions of Neededness and Essentiality

We define P-*needed*, P-*essential*, and P-*erased* redexes, for any reduction P, and study their (relative) properties.

Definition 3.1 • Let $P : t \twoheadrightarrow$ and $u \in t$, in a DRS. We call u *erased* in P or P-*erased* if $u/P = \emptyset$, or equivalently, if $u/Q = \emptyset$ for any $Q \approx_L P$. We say that P *discards* u if P is external to u and erases it [GlKh94a].

• We call u P-*needed*, written $NE_P(u, t)$, if there is no $Q \approx_L P$ that is external to u, and call it P-*unneeded*, $UN_P(u, t)$, otherwise. We call u P-*essential*, $ES_P(u, t)$, if there is no $Q \approx_L P$ that discards u, and call it P-*inessential*, $IE_P(u, t)$, otherwise.

We extend these concepts to reductions co-initial with those containing u as a redex of one of its terms.

• Let $Q : t \twoheadrightarrow$, $P : t \overset{P'}{\twoheadrightarrow} s \twoheadrightarrow$, and $u \in s$. We say $NE_Q(u, s)$ if $NE_{Q/P'}(u, s)$. We call P Q-*needed* if so is every redex contracted in P. We call P *self-needed* if it is P-needed. The other concepts above are extended in the same way.

Clearly, P-neededness, erasure, and essentiality do not depend on the choice of a reduction in $\langle P \rangle_L = \{Q \mid Q \approx_L P\}$. Note that P-needed redexes need not be P-erased in a DRS with a duplicating residual relation. We show this using an OTRS, an example of a DRS, defined using a rule set $R = \{f(x) \to h(x, x), g(x) \to x\}$, with the usual residual relation, and using subterms to label redexes. Consider: $P : t = f(g(x)) \to h(g(x), g(x)) \to h(x, g(x))$ Here $g(x) \in t$ is P-needed, but not P-erased. Note also that P-essential redexes need not be P-needed. For: $P : u = f(g(x)) \to f(x)$ in $R = \{f(x) \to x,\, g(x) \to x\}$, redex u is not P-needed, but it is P-essential. However, for the redexes contracted in P (not the redexes whose residuals are contracted in P), P-neededness and P-essentiality do coincide:

Lemma 3.1 Let $u \in t \stackrel{P}{\twoheadrightarrow}$, in a DRS.

(1) If u is P-needed, then it is P-essential, but the converse need not hold.

(2) If u is P-erased and P-essential, then it is P-needed.

(3) If P contracts a redex v, then v is P-needed iff it is P-essential.

Proof. Easy consequence of Definition 3.1. (Or see [KhGl96].)

Corollary 3.1 A reduction P in a DRS is self-needed iff it is self-essential.

3.2 Properties of Discrete Neededness and Essentiality

We now embark on proving existence of P-needed redexes for any finite $P : t \twoheadrightarrow s$ in a stable DRS. The idea of our proof is simple, and resembles that for normalizing reductions [GlKh96]. We show that the last step of P is P-needed, and that any P-needed redex in a non-initial term of P is either a residual of such a redex in the preceding term, or is created by contraction of a P-needed redex; therefore the redex in t 'responsible' for the last step of P is P-needed.

Proposition 3.1 Let $P : s \twoheadrightarrow t \stackrel{u}{\to} e \stackrel{P^*}{\twoheadrightarrow}$, in a stable DRS.

(1) Let u create $v \in e$, and let $UN_P(u, t)$. Then so is v.

(2) Let $UN_P(w, t)$ and let $v \in e$ be its u-residual. Then $UN_P(v, e)$.

(3) Let $NE_P(v, t)$, and let $v \neq u$. Then v has at least one u-residual v', and if it is the only u-residual of v, then $NE_P(v', e)$.

Proof. (1) By Lemma 3.1.(3), $UN_P(u, t) \Rightarrow IE_P(u, t)$. Hence there is $Q_t \approx_L u + P^*$ that discards u. By Lemma 2.2, $Q_e = Q_t/u$ is external to v. Further, $Q_e \approx_L P^*$. Hence $UN_P(v, e)$.

$$\begin{tikzcd}
s \arrow[r, two heads] & t \arrow[r, "u"] \arrow[d, "Q_t"'] & e \arrow[r, two heads, "P^*"] \arrow[d, "Q_e"] & P \\
 & {} & {} &
\end{tikzcd}$$

(2) Since w is P-unneeded, there is $Q_t \approx_L u + P^*$ that is external to w (see the picture for (1)). Hence $u/Q_t = \emptyset$ (since $u/(u+P^*) = \emptyset$). Further, $Q_e = Q_t/u \approx_L P^*$. By Lemma 2.1 Q_e is external to v. Thus $UN_P(v, e)$.

(3) If on the contrary $v/u = \emptyset$, then $u + P^*$ would discard v, contrary to $NE_P(v, t)$. If v' is the only u-residual of v then if v' was P^*-unneeded, there would be $Q_e \approx_L P^*$ that is external to v'. Then $u + Q_e$ would be external to v, contradiction.

The above proposition, as well as Lemma 5.3, is valid for P-(un)neededness replaced by P-(in)essentiality. We omit proofs (to be found in [KhGl96]) since the corresponding statements are not needed for the main results of this paper.

Lemma 3.2 Let $P : t \twoheadrightarrow s \xrightarrow{u} o$, in a DRS. Then u is P-needed.

Proof. By Lemma 2.3, there is no $Q : s \twoheadrightarrow o$ such that Q is external to u and $Q \approx_L u$. Hence u is u-needed, i.e., P-needed.

Theorem 3.1 Let $P : t \twoheadrightarrow s$ be a non-empty finite reduction in a stable DRS. Then t contains a P-needed redex.

Proof. Let $P : t = t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots \xrightarrow{u_{n-1}} t_n = s$, and let $v_0, \ldots, v_{n-1}$ be such that $v_i \in t_i$, $v_{n-1} = u_{n-1}$, and either $v_i \neq u_i$ and v_{i+1} is a u_i-residual of v_i, or $v_i = u_i$ and u_i creates v_{i+1}. By Lemma 3.2, v_{i-1} is P-needed, and so are all v_i by Proposition 3.1.(1)-(2). Thus $v_0 \in t$ is P-needed.

4 Standardization

In orthogonal acyclic 'syntactic' rewrite systems, if a redex u erases or duplicates another redex v, then v is 'below' u, and v cannot duplicate or erase u. Therefore, if a reduction $P : t \twoheadrightarrow s$ discards a redex $u \in t$, then one easily constructs a $Q \approx_L P$ that discards u and does not contract redexes inside u (see e.g., [GlKh94]), and clearly such a Q discards all redexes in the arguments of u. In DRSs, where we do not have a nesting relation on redexes, the above property can be approximated by saying that 'P-inessential redexes cannot duplicate P-essential ones' (obviously, P-inessential redexes cannot erase P-essential ones). We will only need the following slightly weaker (by Lemma 3.1) property:

Definition 4.1 We call $P : t \twoheadrightarrow$, or equivalently $\langle P \rangle_L$, *regular* or *semi-linear* if, for any $Q \approx_L P$, P-unneeded steps of Q do not duplicate P-needed redexes.

Note that our [acyclicity] axiom is too weak to exclude irregular reductions from SDRSs. Consider simply the ARS given by the graph of a term t containing two redexes $t \xrightarrow{u} o$ and $t \xrightarrow{v} s$, such that $v/u = \emptyset, u/v = \{u_1, u_2\}$, and contraction of u_1 and u_2 in either order yields o. Then, for $P = u$, $NE_P(u,t)$, $UN_P(v,t)$, but v duplicates u.

Lemma 4.1 Let P be regular and let $Q \trianglelefteq P$, in a DRs. Then P/Q is regular.

Proof. Immediate, since for any redex in $\langle P/Q \rangle_L$, P- and P/Q-neededness coincide.

Theorem 4.1 Let $P : t \twoheadrightarrow s$ be a non-empty regular reduction in a stable DRS. Then t contains a P-erased P-needed redex; for example, the P-needed redex in t whose residual is contracted first among residuals of P-needed redexes in t.

Proof. Let $P : t = t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots \to t_n = s$, and let u_i be the first among the contracted residuals of P-needed redexes in t_0. By Theorem 3.1, t_0 contains a P-needed redex, and at least one residual of each P-needed redex in t_0 is contracted in P; thus the redex u_i as above exists. Let u_i be the residual of $v \in t_0$. By Proposition 3.1.(1)-(2), every u_j with $j < i$ is P-unneeded. By regularity of P and Proposition 3.1.(3), u_i is the only residual of v in t_i, and is P-needed. Hence v is P-erased (and P-needed).

Note that for any finite $P : t \twoheadrightarrow s$, P-needed redexes all of whose residuals along P remain P-needed are P-erased, since s does not contain $\emptyset$-needed redexes.

Definition 4.2 (1) Let $P : t \twoheadrightarrow$ and $Q \trianglelefteq P$. The *(canonical) P-needed variant of* Q, written $SE_P(Q)$, is defined as follows: let $v \in t$ be such that it is P-needed and its residual is contracted in Q first among P-needed residuals of P-needed redexes in t. Then $SE_P(Q) = v + SE_{P/v}(Q/v)$. If there is no such a redex in t, then $SE_P(Q) = \emptyset$.
(2) We call $SE_P(P)$ the *self-needed variant of* P and denote it by $SE(P)$.

Lemma 4.2 Let $Q \trianglelefteq P$ and P be regular. Then $SE_P(Q)$ is a P-erased P-needed reduction; its length coincides with the number of P-needed steps in Q; and $SE_P(Q) \trianglelefteq Q \trianglelefteq P$.

Proof. Let $Q : t_0 \overset{Q_i}{\twoheadrightarrow} t_i \xrightarrow{u_i} t_{i+1} \overset{Q^*_{i+1}}{\twoheadrightarrow}$, and let $u_{i_0}, u_{i_1}, \ldots$ be all the P-needed steps in Q ($i_0 < i_1 < \ldots$). Since u_{i_0} is the first P-needed step in Q, it follows from Proposition 3.1.(1)-(2) that u_{i_0} is a residual of some P-needed redex $v_0 \in t_0$, and u_{i_0} is the only residual of v_0 by regularity of P.[3] Hence v_0 is P-erased. Thus $SE_P(Q) = v_0 + SE_{P/v_0}(Q/v_0)$ where v_0 is P-needed and P-erased. Similarly, we show that $SE_P(Q) : t_0 = s_0 \xrightarrow{v_0} s_1 \xrightarrow{v_1} s_2 \twoheadrightarrow$, where u_{i_j} is the first P'_j-needed step in $Q'_j = Q/(v_0 + v_1 + \ldots + v_{j-1})$, where $P'_j = P/(v_0 + v_1 + \ldots + v_{j-1})$, and u_{i_j} is the only residual of P-erased P-needed redex $v_j \in s_j$ along Q'_j (see the figure). (Note that $P'_j \approx_L P$, so P'_j-neededness coincides with P-neededness.) Hence $SE_P(Q) \trianglelefteq Q$.

$$\begin{tikzcd}
t_0 = s_0 \arrow[r, two heads, "Q_{i_j}"] \arrow[d, two heads] & t_{i_j} \arrow[r, "u_{i_j}"] \arrow[d, two heads, "\emptyset"] & t_{i_j+1} \arrow[r, two heads, "Q^*_{i_j+1}"] & Q'_0 = Q \\
s_j \arrow[r, two heads] \arrow[d, "v_j"'] & t_{i_j} \arrow[r, "u_{i_j}"] \arrow[d, "u_{i_j}"] & t_{i_j+1} \arrow[r, two heads, "Q^*_{i_j+1}"] \arrow[d, two heads, "\emptyset"] & Q'_j \\
s_{j+1} \arrow[r, two heads] & t_{i_j} \arrow[r, two heads, "\emptyset"] & t_{i_j+1} \arrow[r, two heads, "Q^*_{i_j+1}"] & Q'_{j+1}
\end{tikzcd}$$

It is easy to see that standard reductions, in the sense of [Bar84, HuLé91, Klo80, GLM92], are self-essential. In fact, the notion of self-essential reduction is the best approximation to the notion of standard reduction for DRSs, and it captures the essence of standardization in many respects (see [KhGl96a] for the use of self-essential standard reductions in an abstract optimality theory). Therefore:

[3] Here we also use the fact that $Q \trianglelefteq P$, so that every finite initial part of Q can be considered as being an initial part of P, up to $\approx_L$, and Proposition 3.1.(1)-(2) applies.

Definition 4.3 We call a reduction in a DRS *standard* if it is self-essential.

Theorem 4.2 (Standardization) For any finite semi-linear reduction P in a stable DRS, $SE(P)$ is a standard finite reduction Lévy-equivalent to P.

Proof. By Lemma 4.2, $SE(P) \trianglelefteq P$ and is finite. Let $P : t_0 \stackrel{P_i}{\twoheadrightarrow} t_i \stackrel{u_i}{\rightarrow} t_{i+1} \twoheadrightarrow t_n$ and $u_{i_0}, \ldots, u_{i_{k-1}}$ be all P-needed steps in P ($i_0 < i_1 < \ldots$). Then, as shown in the proof of Lemma 4.2, $SE(P) : t_0 = s_0 \stackrel{v_0}{\rightarrow} s_1 \stackrel{v_1}{\rightarrow} \ldots \stackrel{v_{k-1}}{\rightarrow} s_k$, where u_{i_j} is the first P'_j-needed step in $P'_j = P/(v_0 + v_1 + \ldots + v_{j-1})$, and u_{i_j} is a residual of v_j along P'_j (see also the diagram for Lemma 4.2; take $Q = P$). Further, s_k does not contain $P/SE(P)$-erased $P/SE(P)$-needed redexes by Definition 4.2. By Lemma 4.1 and Lemma 4.2, $P/SE(P)$ is regular. By Theorem 4.1, $P/SE(P) = \emptyset$, i.e., $P \approx_L SE(P)$. Hence $SE(P)$ is self-needed by Lemma 4.2, thus self-essential by Corollary 3.1.

The following example shows that if P is infinite, $P \approx_L SE(P)$ need not hold. Consider the OTRS $R = \{f(x) \rightarrow g(h(x)), g(x) \rightarrow f(h(x)), h(x) \rightarrow a\}$ and the reductions $P : f(x) \rightarrow g(h(x)) \rightarrow g(a) \rightarrow f(h(a)) \rightarrow f(a) \rightarrow g(h(a)) \rightarrow g(a) \rightarrow \ldots$ and $Q : f(x) \rightarrow g(h(x)) \rightarrow f(h(h(x))) \rightarrow g(h(h(h(x)))) \rightarrow \ldots$. Then it is easy to check that $Q = SE(P)$, it is the only P-essential reduction, but $P/Q \neq \emptyset$.

This should not be understood as implying that needed reductions are not useful in the study of transfinite reductions in OTRSs [KKSV95] – needed-fair reductions *strongly converge* to infinite normal forms. The reduction P above is divergent. In [KKSV95], reductions that do not converge strongly are considered meaningless, while here we study normalization relative to *any* reduction.

5 Discrete Normalization

This section is devoted to proving the main result of the paper – the Discrete Normalization Theorem. First a simple lemma, which is immediate from Definition 3.1.

Lemma 5.1 Let $P, Q : t \twoheadrightarrow$, in a DRS. Then Q is P-erased iff $Q \trianglelefteq P$.

Definition 5.1 Let $P, Q : t \twoheadrightarrow$. We call Q *P-fair* if Q contracts only P-erased redexes, and every P-erased redex in any term of Q is erased in Q as well.

Lemma 5.2 Let $P, Q : t \twoheadrightarrow$, let $P = u + P'$, let $Q' = Q/u$, and let Q be P-fair. Then Q' is P'-fair (see the diagram).

Proof. Obviously Q is P-fair implies Q is P-erased. Further, $P \trianglelefteq Q$ implies $P' \trianglelefteq Q'$. Hence P' is Q'-erased by Lemma 5.1. It remains to show that any P'-erased redex w' in some term of Q' is erased in Q' too. Since u is P-erased, it is Q-erased by the assumption (Q is P-fair). So let $Q^* \leq Q$ be finite and erase u, let $Q = Q^* + Q_1^*$, and let $Q'^* = Q^*/u$; hence $Q' = Q'^* + Q_1^*$. We can assume that Q^* is chosen so that w' is in some term of Q'^*. Let W' be the set of residuals of w' in the final term o of Q'^*. Since w' is P'-erased, so is W' (i.e., $W'/(P'/Q'^*) = \emptyset$). Since $P/Q^* \approx_L P'/Q'^*$, W' is P-erased too. Since Q is P-fair by the assumption, W' is erased in Q as well, i.e., $W'/Q_1^* = \emptyset$. Hence W' is erased in Q', and so is w'.

$$\begin{tikzcd}
{} \arrow[r, "u"] \arrow[d, two heads, "Q^*"'] & {} \arrow[r, two heads, "P'"] \arrow[d, two heads, "Q'^* \ni w'"] & P \\
o \arrow[r, two heads, "\emptyset"] \arrow[d, two heads, "Q_1^*"'] & o \supseteq W' \arrow[d, two heads, "Q_1^*"] & \\
Q & Q' &
\end{tikzcd}$$

Lemma 5.3 Let $P : t \xrightarrow{u} t' \xrightarrow{P'}\!\!\!\!\rightarrow s$ be regular, let $NE_P(v, t)$, and let $u \neq v$. Then v has at least one P-needed u-residual in t'.

Proof. Let $v/u = \{v_1, \ldots, v_k\}$, and suppose on the contrary that $UN_P(v_i, P')$ for all $i = 1, \ldots, k$ ($k \geq 1$ by Proposition 3.1.(3)). Thus, for every i, there is $P_i \approx_L P'$ that is external to v_i. P' is regular by Lemma 4.1. Hence, by Lemma 4.2, $SE(P')$ is P'-needed, and by Proposition 3.1.(2) all residuals of v_i along $SE(P')$ are P'-unneeded. Hence $SE(P')$ is external to all v_i, and $u + SE(P')$ is external to v, contradiction, since $P \approx_L u + SE(P')$ by the Standardization theorem, and $NE_P(v, t)$.

Lemma 5.4 Let $P : t \xrightarrow{u} e \twoheadrightarrow s$ be regular, $Q : t \twoheadrightarrow \infty$ P-erased and P-needed. Then $u + Q/u \trianglelefteq P$ and Q/u contracts infinitely many P-erased P-needed redexes.

Proof. By Lemma 5.1, $Q \trianglelefteq P$, hence $u + Q/u \trianglelefteq P$. Let $Q : t = t_0 \xrightarrow{v_0} t_1 \xrightarrow{v_1} \ldots$ and $Q_i : t = t_0 \xrightarrow{v_0} t_1 \xrightarrow{v_1} \ldots \rightarrow t_i$. If $v_i \notin u/Q_i$, then by Lemma 5.3 v_i has at least one P-needed u/Q_i-residual, and $v_i/(u/Q_i)$ contracts at least one P-needed redex. Suppose on the contrary that there are only a finite number of such i. Then there is a k such that $t_k \xrightarrow{v_k} t_{k+1} \xrightarrow{v_{k+1}} \ldots$ is an (infinite) u/Q_k-development, contradiction.

Theorem 5.1 (Discrete Normalization) Let $P : t \twoheadrightarrow$, in a stable DRS.

(1) $P \approx_L Q$ iff Q is P-fair. In particular, any multi-step reduction $Q : t = t_0 \twoheadrightarrow t_1 \twoheadrightarrow \ldots$ that contracts in parallel all P-erased redexes in every t_i is Lévy-equivalent to P.

(2) If P is finite and semi-linear, then any reduction $t \twoheadrightarrow$ that contracts P-erased P-needed redexes eventually terminates; the resulting reduction is Lévy-equivalent to P, even if a finite number of P-erased P-unneeded redexes, and only such, are also contracted.

Proof. (1) ($\Rightarrow$) Let $P \approx_L Q$. Then $Q/P = \emptyset$, and Q is P-erased by Lemma 5.1. For any $Q' < Q$, $P/Q' \approx_L Q''$, where $Q : t \xrightarrow{Q'}\!\!\!\!\rightarrow s \xrightarrow{Q''}\!\!\!\!\rightarrow$. Hence any P-erased redex $u \in s$ is Q-erased too, i.e., Q is P-fair. ($\Leftarrow$) $Q/P = \emptyset$ follows from Lemma 5.1. So we have to show that $P/Q = \emptyset$. Let $P = u + P'$. Then since every P-erased redex of Q is erased in Q, we have that $u/Q = \emptyset$. By Lemma 5.2, $Q' = Q/u$ is P'-fair. Hence, we can show similarly that if $P' = u' + P''$, then $u'/Q' = \emptyset$, and so on. Thus $P/Q = \emptyset$.

(2) Let $P : t \xrightarrow{u} o \twoheadrightarrow s$. Suppose there is a P-erased reduction $Q : t \twoheadrightarrow$ that contracts infinitely many P-needed redexes. By Lemma 4.2 $SE_P(Q)$ is an infinite P-needed P-erased reduction, and by Lemma 5.4, $SE_P(Q)/u$ is a P-erased reduction with infinitely many P-needed steps, starting from o. Similarly, one shows that there is an infinite P-erased P-needed, i.e., $\emptyset$-erased $\emptyset$-needed, reduction starting from s, contradiction.

The DN theorem implies that $\langle P\rangle_L$ contains only a finite number of standard reductions, for any finite semi-linear P. It also implies termination of known standardization algorithms (which transform finite reductions into standard ones) [Bar84, HuLé91, Klo80, GLM92, Oos96], since our standardization concept is the most liberal one known in the literature.

Let us call a reduction P *self-neededly fair* if no redex in a term of P has a P-needed P-erased residual in every subsequent term. Then it follows from the example demonstrating failure of standardization for infinite reductions that even self-neededly fair P-needed P-erased reductions need not be Lévy-equivalent to P. So (1) of Theorem 5.1 cannot be improved by allowing self-neededly fair sequential reductions instead of multi-step reductions.

Finally, note that, in a non-duplicating SDRS, for any finite semi-linear P, $SE(P)$ a shortest reduction in $\langle P\rangle_L$. To show this, let $SE(P) = u + P'$, $Q \approx_L P$, and assume that P' is shortest in $\langle P'\rangle_L$. Then $|Q/u| \leq |Q| - 1$ since exactly one residual of u is contracted in Q, and $|SE(P)| \leq |Q|$ follows from the induction assumption $|P'| \leq |Q/u|$.

6 Denotational Semantics

Lisper [Lis96] defined semantics for an ARS $A = (Ter, Red, \to)$[4] as follows: let $\mathcal{C} = (C, \sqsubseteq, \perp)$ be a cpo and let $f : Ter \to C$ be a *monotone* function, meaning that $t \to s$ implies $f(t) \sqsubseteq f(s)$. Then *semantics* of a term $t \in Ter$ w.r.t. f in $\mathcal{C}$, $S(t, f, \mathcal{C})$, is the set of least upper bounds $\sqcup f(P)$, such that there is no co-initial Q such that $P <_m Q$ and $\sqcup f(P) \sqsubset \sqcup f(Q)$, where, for any $P : t \to t_1 \to t_2 \to \ldots$, $f(P) = f(t), f(t_1), f(t_2), \ldots$, and $\sqcup f(P)$ denotes the least upper bound of $f(P)$ in $\mathcal{C}$; and $P <_m Q$ means that every term of P can be reduced to a term in Q. Elements of $S(t, f, \mathcal{C})$ are *values* of t in $\mathcal{C}$, w.r.t. f.

The most commonly used cpos as semantics for syntactic rewrite systems are those of *tree* cpos $\mathcal{T} = (T_\perp^\infty, \preceq, \perp)$, where $T_\perp^\infty$ consists of finite and infinite trees labelled by symbols from the rewrite alphabet and the distinguished constant symbol $\perp$ which represents *undefined* or *meaningless* terms; and $\preceq$ is the ordering induced by $\perp \preceq s$ for any $s \in T_\perp^\infty$. The tree cpo was first defined for Recursive Program Schemes by Nivat [Niv75], and was later extended to the λ-calculus by Lévy [Lév76] (the so called Lévy-Longo trees or Lazy-trees [Lon83, AbOn93]) and Barendregt [Bar84] (the *Böhm* trees); and to TRSs by Boudol [Bou85]. In Nivat's and Boudol's semantics, undefined are terms which cannot be reduced to a *root stable form* – a term that cannot be reduced to a redex. In Böhm semantics, λ-terms not having a head-normal form (hnf) are considered meaningless, and Lévy defines meaningless to be λ-terms that do not have neither hnfs nor weak hnfs (which are simply abstractions λxt). Reasonable concepts of meaningless terms in (infinitary) TRSs and the λ-calculus are studied in [KOV96].

Lisper's definition of abstract semantics allows one to describe all the above tree semantics, but is not suitable for all known interpretations, and we generalize it slightly by taking f to be a function from the set of all reductions in A to C. Such

[4] More precisely, ARSs used in [Lis96] are Klop's ARSs [Klo92].

an f is *monotone* if for any reduction $P : t \overset{P'}{\twoheadrightarrow} s \overset{u}{\rightarrow} o$ in A, $f(P') \sqsubseteq f(P)$ in $\mathcal{C}$. So the information increases (w.r.t. $\sqsubseteq$, in $\mathcal{C}$) as the computation progresses, and the whole history of the computation is taken into account. We define every $\sqsubseteq$-maximal element in the set $\{\sqcup f(P) \mid P : t \twoheadrightarrow \}$ (which need not be $\sqsubseteq$-maximal in $\mathcal{C}$) to be a *value* of t. We will call this semantics the *maximal* semantics.

Lévy's reduction space $\mathcal{L} = (L, \trianglelefteq)$, where L is the set of all Lévy-equivalence classes of reductions in a DRS R, is also a cpo (the bottom element is irrelevant). And the interpretation f on reductions in R defined by $f(P) = \langle P \rangle_L$ is obviously monotone. If we take $R = (A, /)$ where terms of $A = (Ter, Red, \rightarrow)$ are (finite) λ-terms with constant $\perp$; Red is the set of β-redex occurrences; and $\rightarrow$ is defined by: $t \overset{u}{\rightarrow}_\beta s$ if $u \in t$ is a β-redex having a hnf, and $t \overset{u}{\rightarrow}_\perp s$ otherwise (where $t \overset{u}{\rightarrow}_\perp s$ means that s is obtained from t by replacing u with $\perp$); and take as the residual relation the one induced by that of the λ-calculus, then the maximal semantics of R in $\mathcal{L}$ (w.r.t. f above) is exactly the Böhm semantics of the λ-calculus. The Nivat, Boudol and Lévy's semantics can be recovered similarly, by taking appropriate R and $\rightarrow_\perp$.

Alternatively, we can define all redexes inside meaningless redexes to be meaningless, and consider only ordinary *meaningful-fair* reductions, i.e., reductions without $\perp$-steps in which no meaningful redex has a meaningful residual in every subsequent term. In all above four cases, there is (exactly one, up to $\approx_L$) $\trianglelefteq$-minimal meaningful-fair reduction starting from a term t (such reductions contract only meaningful redexes), and these reductions can be taken as representations of corresponding semantic trees.

Note that, unlike the maximal semantics, not-necessarily fair reductions are used to define denotational semantics in the latter case. And the DN theorem gives a characterization of reductions computing the values of terms in all the above cases.

7 Conclusions and Future Work

We have proven an abstract Discrete Normalization theorem. It gives a strategy for constructing reductions Lévy-equivalent to a finite or infinite reduction P (which need not be normalizing or fair). We introduced a number of concepts and constructs, such as standardization, which are technically interesting in their own rights.

We have not addressed here the question of how to find efficiently discretely needed redexes, and a syntactic theory similar to the strong sequentiality theory [HuLé91, KlMi90] needs to be developed for discrete and relative normalization.

One could try to develop a theory of normalization relative to *regular stable* sets Π of reductions which would unify discrete normalisation with the finitary relative normalisation of [GlKh96]. Thus by taking the set of all $\mathcal{S}$-normalizing reductions for regular stable sets $\mathcal{S}$ of final terms, or taking a Lévy-equivalence class of regular reductions for Π, one would get the RN and DN theorems, respectively.

Finally, development of a needed narrowing theory in [AEH94] suggests that the abstract theory of neededness can be extended to cover narrowing and possibly other proof procedures. Non-deterministic systems must also be considered, and Boudol's work [Bou85] extending [HuLé91] to non-orthogonal TRSs can be used as the basis.

Acknowledgments We thank J.R. Kennaway, V. Van Oostrom and M.R. Sleep for useful comments. The diagrams were drawn using P. Taylor's diagram package.

References

[AbOn93] Abramsky S., Ong C.-H. L. Full abstraction in the lazy lambda calculus. Inf.& Comp., 105:159-267, 1993.

[AEH94] Antoy S., Echahed R., Hanus M. A needed narrowing strategy. In: Proc. of POPL'94, Portland, Oregon, 1994.

[AnMi94] Antoy S. and Middeldorp A. A Sequential Reduction Strategy. In Proc. of ALP'94, Springer LNCS, vol. 850, p. 168-185, 1994.

[AsLa96] Asperti A., Laneve C. Interaction Systems II: the practice of optimal reductions. TCS, 159(2):191-244, 1996.

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

[BKKS87] Barendregt H. P., Kennaway J. R., Klop J. W., Sleep M. R. Needed Reduction and spine strategies for the lambda calculus. Inf.& Comp., 75(3):191-231, 1987.

[BeLé79] Berry G., Lévy J.-J. Minimal and optimal computations of recursive programs. JACM 26, 1979, p. 148-175.

[Bou85] Boudol G. Computational semantics of term rewriting systems. In: Algebraic methods in semantics. Nivat M., Reynolds J.C., eds. Cambr. Univ. Press, 1985, p. 169-236.

[CuFe58] Curry H. B., Feys R. Combinatory Logic. vol. 1, North-Holland, 1958.

[Gar94] Gardner P. Discovering needed reductions using type theory. In: Proc. of TACS'94, Springer LNCS, v. 789, M. Hagiya, J. C. Mitchell, eds. Sendai, 1994, p. 555-574.

[GlKh94] Glauert J.R.W., Khasidashvili Z. Relative Normalization in Orthogonal Expression Reduction Systems. In: Proc. of CTRS'94, Springer LNCS, vol. 968, N. Dershowitz, N. Lindenstrauss, eds. Jerusalem, 1994, p. 144-165.

[GlKh94a] Glauert J.R.W., Khasidashvili Z. Minimal and optimal relative normalization in Expression Reduction Systems. Report SYS-C94-06, UEA, Norwich, 1994.

[GlKh96] Glauert J.R.W., Khasidashvili Z. Relative normalization in deterministic residual structures. In [CAAP'96], p. 180-195.

[GlKh96a] Glauert J.R.W., Khasidashvili Z. Relative normalization in stable deterministic residual structures. Report SYS-C96-01, UEA, Norwich, 1996.

[GLM92] Gonthier G., Lévy J.-J., Melliès P.-A. An abstract Standardisation theorem. In: Proc. of LICS'92, IEEE Computer Society Press, 1992, p. 72-81.

[Hin64] Hindley R.J. An abstract form of the Church-Rosser theorem I. JSL, 34(4):545-560, 1969.

[HuLé91] 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.

[Kat90] Kathail V. Optimal Interpreters for Lambda-calculus based functional languages, PhD thesis, MIT, 1990.

[Ken89] Kennaway J.R. Sequential evaluation strategy for parallel-or and related reduction systems. Annals of Pure and Applied Logic, 43:31-56, 1989.

[KeSl89] Kennaway J. R., Sleep M. R. Neededness is hypernormalizing in regular combinatory reduction systems. Preprint, University of East Anglia, Norwich, 1989.

[KKSV95] Kennaway J. R., Klop J. W., Sleep M. R, de Vries F.-J. Transfinite reductions in orthogonal term rewriting. Information and Computation, 119(1):18-38, 1995.

[KOV96] Kennaway J.R., van Oostrom V., de Vries F.J. Meaningless terms in rewriting. In: Proc. of the 5^{th} International Conference on Algebraic and Logic Programming, ALP'96, Aachen, 1996. To Appear.

[Kha88] Khasidashvili Z. β-reductions and β-developments of λ-terms with the least number of steps. In: Proc. of the Int. Conference on Computer Logic COLOG'88, Tallinn 1988, Springer LNCS, v. 417, P. Martin-Löf and G. Mints, eds. 1990, p. 105-111.

[Kha92] Khasidashvili Z. The Church-Rosser theorem in Orthogonal Combinatory Reduction Systems. Report 1825, INRIA Rocquencourt, 1992.
[Kha93] Khasidashvili Z. Optimal normalization in orthogonal term rewriting systems. In: Proc. of RTA'93, Springer LNCS, vol. 690, C. Kirchner, ed. Montreal, 1993, p. 243-258.
[Kha94] Khasidashvili Z. On higher order recursive program schemes. In: Proc. of CAAP'94, Springer LNCS, vol. 787, S. Tison, ed. Edinburgh, 1994, p. 172-186.
[KhGl96] Khasidashvili Z., Glauert J.R.W. Discrete normalization and standardization in stable deterministic residual structures. Report SYS-C96-02, UEA, Norwich, 1996.
[KhGl96a] Khasidashvili Z., Glauert J.R.W. Zig-zag and extraction families in non-duplicating stable Deterministic Residual Structures. Submitted for publication. (Available at ftp://ftp.sys.uea.ac.uk/pub/zurab/)
[CAAP'96] Kirchner H., editor. Proc. of the 19^{th} International Colloquium on Trees in Algebra and Programming, CAAP'96, Springer LNCS, vol. 1059, 1996.
[Klo80] Klop J. W. Combinatory Reduction Systems. Mathematical Centre Tracts n. 127, Amsterdam, 1980.
[Klo92] Klop J. W. Term Rewriting Systems. In: S. Abramsky, D. Gabbay, and T. Maibaum eds. Handbook of Logic in Comp. Science, vol. II, Oxford Univ. Press, 1992, p. 1-116.
[KlMi90] Klop J.W., Middeldorp A. Sequentiality in Orthogonal Term Rewriting Systems. Journal of Symbolic Computation 12:161-195, 1991.
[Lam90] Lamping J. An algorithm for optimal lambda calculus reduction. In: Proc. of POPL'90, 1990, p.16-30.
[Lév76] Lévy J.-J. An algebraic interpretation of the $\lambda\beta K$-calculus; and an application of a labelled λ-calculus. TCS 2(1):97-114, 1976.
[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.
[Lis96] Lisper B. Computing in unpredictable environments: semantics, reduction strategies, and program transformations. In [CAAP'96], p. 165-179.
[Lon83] Longo G. Set theoretic models of lambda calculus: theories, expansions and isomorphisms. Annals of pure and applied logic, 24:153-188, 1983.
[Mar92] Maranget L. La stratégie paresseuse. Thèse de l'Université de Paris VII, 1992.
[Mel96] Melliès P.-A. Description Abstraite des Systèmes de Réécriture. Thèse de l'Université Paris 7, 1996.
[Nip93] Nipkow T. Orthogonal higher-order rewrite systems are confluent. In: Proc. of TLCA'93, Springer LNCS, vol. 664, Bazem M., Groote J.F., eds. 1993, p. 306-317.
[Niv75] Nivat M. On the interpretation of recursive polyadic program schemes. Symposia Mathematica, 15:255-281, 1975.
[Nök94] Nöcker E. Efficient Functional Programming. Compilation and Programming Techniques. Ph.D. Thesis, Katholic University of Nijmegen, 1994.
[Oos94] Van Oostrom V. Confluence for Abstract and Higher-Order Rewriting. Ph.D. Thesis, Free University of Amsterdam, 1994.
[Oos96] Van Oostrom V. Higher order families. In: Proc. of RTA'96. To appear.
[Raa96] Van Raamsdonk F. Confluence and normalisation for higher-order rewriting. Ph.D. Thesis, CWI Amsterdam, 1996.
[SeRa93] Sekar R.C., Ramakrishnan I.V. Programming in Equational Logic: Beyond Strong Sequentiality. Information and Computation, 104(1):78-109, 1993.
[SPE93] Sleep M. R., Plasmeijer M. J., van Eekelen M. C. J. D., eds. Term Graph Rewriting: Theory and Practice. John Wiley, 1993, p.141-156.
[Sta89] Stark E. W. Concurrent transition systems. TCS, 64(3):221-270, 1989.