An Abstract Theory of Optimal Evaluation *

Zurab Khasidashvili

NTT Basic Research Laboratories
Atsugi, Kanagawa, 243-01, Japan
zurab@theory.brl.ntt.co.jp

John Glauert

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

Abstract

This paper generalizes the Huet and Lévy theory of normalization by neededness and Lévy's optimality theory to an abstract setting. We define *Stable Deterministic Residual Structures* (SDRS) and *Deterministic Family Structures* (DFS) by axiomatizing the *residual* and *family* relations in an *Abstract Reduction System*. We present two proofs of the *Relative Normalization Theorem*, one for SDRSs for *regular stable* sets, and another for DFSs for all *stable* sets of desirable 'normal forms', and prove a Relative Optimality Theorem for DFSs. We further introduce and study a concept of *implementation* of DFSs. We show that for any DFS, its implementation is a non-duplicating DFS with zig-zag as the family relation, where zig-zag is simply the symmetric and transitive closure of the residual relation on redexes with histories. Optimal family-reductions in a DFS become needed reductions in the non-duplicating implementation DFS. We show that *sharing* formalized by our concept of a family is compositional: the sharing in a DFS can be decomposed into a weaker sharing, such as zig-zag, and a sharing in the implementation of the latter. These results arise from a study of the family relation in non-duplicating SDRSs. We show that zig-zag forms a family-relation in every non-duplicating SDRS, and that it is the only *separable* family relation in such SDRSs. To prove this, we develop an abstract *extraction* procedure, and to show that the family concepts defined via zig-zag and via extraction yield the same relation.

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

1 Introduction

In order to achieve optimal evaluation of λ-terms, Lévy introduced a notion of *redex-family* to capture the concept of redexes of the 'same origin'. The hope was that it would be possible to mimic multi-step reductions, which contract whole families in a term, by reduction of some graph representation, in which every multi-step would be represented by contraction of a single graph redex [50,51]. It is necessary to consider graph representations since Barendregt et al [9] showed that there exists no one-step optimal recursive β-reduction strategy on λ-terms. An optimal graph implementation has indeed been achieved by Lamping [49] and Kathail [30]. Maranget [52] generalized Lévy's optimality theory to orthogonal Term Rewriting Systems (TRSs), Gonthier et al [23] simplified Lamping's technique, and Asperti and Laneve generalized both Lévy's optimality theory and Gonthier's implementation to Interaction Systems, which cover most of the languages with a constructor-destructor discipline [4,5]. More recently, van Oostrom generalized the optimality theory to the whole class of orthogonal Higher-Order Rewriting Systems (HORSs) [62], and Guerrini developed a general theory of sharing-graph rewriting [25].

Lévy introduced the family concept in three different ways: via an appropriate notion of *labelling*, via *extraction*, and by *zig-zag*. He showed that they all yield the same concept, in the λ-calculus. The same holds for all orthogonal HORSs R if all three family concepts are defined in the refinement of R which decomposes every original R-step into first-order, or TRS, step and a number of substitution steps [62]. However, the zig-zag family can be defined directly in R, and this yields a different, and weaker, family concept [4]. Independently, though much earlier, Kennaway and Sleep [32] defined a concept of labelling for orthogonal Combinatory Reduction Systems (CRSs), which covers orthogonal TRSs and Interaction Systems, improving Klop's original labelling system for CRSs [46]. Their labelling is different from both Maranget's labelling for orthogonal TRSs [52] and Asperti-Laneve's labelling for Interaction Systems [4].

In addition to this variety of family concepts, alternative graph rewriting algorithms have been developed for optimal implementation of orthogonal rewriting systems, such as Term Graph Rewriting [33], Jungle Rewriting [27], and many others, inspired by Wadsworth's pioneering work on graph-based implementation of the λ-calculus [69]. The challenge is to develop an abstract notion of family general enough to cover all the existing notions, but refined enough to support proofs of normalization and optimality results. Usefulness and feasibility of such a theory was pointed out early on by Berry and Lévy in [12]: 'Thus a unifying or axiomatized point of view should be adopted to preserve the syntactic results of this paper. This seems possible along the lines of O'Donnell [60] and Rosen [66]. Furthermore, the axiomatized approach should include the λ-calculus where much of the syntactic part of this paper is valid'.

This work is the first step towards developing an axiomatic theory of optimality. We define a *Deterministic Family Structure* (DFS) as a *Deterministic Residual Structure* (DRS) with axiomatized *family* and *contribution* relations. The family relation in a DFS is simply an equivalence relation containing zig-zag, and the *contribution* relation expresses causal dependencies among families and is formalized in terms of creation of redexes. All existing family relation definitions in the literature [50,46,32,53,4,62] satisfy our DFS axioms. A DRS in turn is an *Abstract Reduction System* (ARS) which has a *residual* relation between redexes in the source and target terms of each transition $t \xrightarrow{u} s$. Redexes of t may be erased by reduction of u, new redexes may be introduced in s, while other redexes of s are considered *residuals* of redexes in t, as specified by the residual relation. Further, the residual relation is generalized to (many-step) reductions, and *permutation-equivalence* on reductions, referred to below as *Lévy-equivalence*, and the *Lévy-embedding relation*, is introduced, as is done for the λ-calculus in [50,51]. Sufficient conditions needed to define permutation equivalence in an abstract setting were stated by Stark [68] (independently from Lévy [50,51], for the case when the residual relation is non-duplicating), and by Gonthier et al. [24].

We first address the problem of normalization in DRSs. A normalizable term in a rewriting system may have an infinite reduction, so it is important to have a *normalizing* strategy which enables one to construct reductions to normal form. For example, it is well known that the leftmost-outermost strategy is normalizing in the λ-calculus [16]. For Orthogonal TRSs, a general normalizing strategy, called the *needed* strategy, was found by Huet and Lévy [29]. The strategy always contracts a *needed* redex – one whose residual has to be contracted in any reduction to normal form. Huet and Lévy showed that any term not in normal form has a needed redex, and that repeated contraction of needed redexes leads to its normal form whenever there is one. (They also defined *strongly sequential* orthogonal TRSs where a needed redex can be computed effectively in every reducible term. We will not discuss this topic here.)

This fundamental work on neededness has been extended in several directions. Barendregt et al. [10], Maranget [53], Nöcker [59] and Middeldorp [57] studied neededness w.r.t. head-normal forms, weak head-normal forms, constructor head-normal forms, and root-stable forms, respectively. Sekar and Ramakrishnan [67] studied normalization via *necessary sets* of redexes. Kennaway et al. [34] studied a needed strategy for infinitary orthogonal TRSs. A different approach to normalization is developed in Kennaway [31] and Antoy and Middeldorp [2]. Antoy et al. [1] designed a needed narrowing strategy. Gardner [17] described a *complete* way of encoding neededness information using a type assignment system.

In [18,22], the present authors addressed the question of normalization relative to a desired set of final terms, considering the properties that a set $\mathcal{S}$ of terms must possess in order for the neededness theory of Huet and Lévy still to make sense. This work is done in the context of orthogonal *Expression Reduction Systems* (ERSs) [36], a form of higher-order rewriting which subsumes Term Rewriting and the λ-calculus. Natural conditions were imposed on $\mathcal{S}$, called *stability*, that are necessary and sufficient for the following *Relative Normalization* (RN) theorem to hold: each $\mathcal{S}$-normalizable term not in $\mathcal{S}$ (that is, not in $\mathcal{S}$*-normal form*) has at least one $\mathcal{S}$-needed redex, and repeated contraction of such redexes always leads to an $\mathcal{S}$-normal form. It is shown also that if a stable $\mathcal{S}$ is *regular*, i.e., if $\mathcal{S}$-unneeded redexes cannot duplicate $\mathcal{S}$-needed ones, then the $\mathcal{S}$-needed strategy is hypernormalizing as well, and *minimal* (w.r.t. the Lévy-embedding relation) $\mathcal{S}$-normalizing reductions exist.

Here we further generalize the theory by abstracting from the structure of terms. We study relative normalization in DRSs. Despite their highly abstract nature, a counterpart of the *stability* property of Berry [11] and Winskel [71] enables us to prove the RN theorem for all *regular* stable sets $\mathcal{S}$. (We actually prove the Relative Hypernormalization theorem.) We show that without this stability axiom the theorem fails. The proof method employed is similar to that in [35,37] (it was discovered by the author independently from [29]), and is based on the fact that $\mathcal{S}$-needed steps in a reduction can be pushed before $\mathcal{S}$-unneeded steps without affecting the number of $\mathcal{S}$-needed steps.

Since for *irregular* stable $\mathcal{S}$, $\mathcal{S}$-unneeded redexes can duplicate $\mathcal{S}$-needed ones, the above proof method does not apply for all stable DRS (SDRSs); for the same reason, the $\mathcal{S}$-needed strategy is no longer hypernormalizing. However, the DFS axioms (in particular, a strong termination axiom [FFD]) ensure that any $\mathcal{S}$-normalizing reduction can be transformed into an $\mathcal{S}$-needed $\mathcal{S}$-normalizing one by pushing $\mathcal{S}$-needed steps forward, and that all $\mathcal{S}$-needed reductions are eventually terminating (at a term in $\mathcal{S}$). Furthermore, for DFSs we show that a strategy that contracts, in an arbitrary order, redexes that belong to $\mathcal{S}$-needed families, but which need not be $\mathcal{S}$-needed themselves, is still $\mathcal{S}$-normalizing. As a corollary, any $\mathcal{S}$*-needed complete family-reduction*, which in any term contracts all members of a family containing an $\mathcal{S}$-needed redex in a multi-step, is eventually $\mathcal{S}$-normalizing. Using the same method as in [51], we show that the latter reductions are also optimal in the sense that they reach $\mathcal{S}$ in a least number of family-reduction steps.

In order to provide an abstract criterion for *correctness* of a graph implementation of complete-family reductions, we introduce an abstract concept of *Lévy-implementation*: With a DFS we associate a *non-duplicating* DRS, called its implementation, whose steps exactly correspond to complete family-reduction steps in the original DFS. For example, implementation DRSs can model sharing-graph implementations of Lévy's complete family-reductions in the λ-calculus (modulo 'book-keeping' steps which in general can be prohibitively expensive [6]). It is not difficult to show that needed (w.r.t. a stable set of results) reductions in the implementation DRS correspond exactly to needed complete family-reductions in the original DFS, implying that the former DRS indeed implements optimal computations in the latter DFS, in the sense of Lévy [50,51].

Further, we show that the family relation in the implementation DRS induced by the original DFS coincides with zig-zag, which is the weakest family sharing (by our definition of families). Note that a family relation in a (non-duplicating or duplicating) DRS need not coincide with zig-zag. An example is Asperti and Laneve's concept of family for Interaction Systems [4], where unlike the λ-calculus a single redex can create multiple members of a family which cannot be related by zig-zag (without employing the internal structure of reduction steps). In non-duplicating DRSs, we call such families *non-separable.*

Actually, we show that the sharing concept formalized in DFSs is *(de)compositional*: any sharing can be decomposed into a weaker sharing and a non-separable sharing in the non-duplicating DRS implementing that weaker sharing. To this end, we investigate the family relation in non-duplicating DFSs. We show that zig-zag forms a family-relation in every non-duplicating SDRS, and that it is the only separable family relation in such SDRSs. These results are obtained by defining an abstract extraction procedure for non-duplicating SDRSs and showing that zig-zag coincides with our extraction-family relation. The technical contribution here is the simplicity of our construction which avoids irrelevant syntactic complications, such as those related to the top-down and left-to right nature of the conventional concept of standardization.

The paper is organized as follows. In the next section, we introduce SDRSs and prove some fundamental lemmas concerning (mutually) *external* reductions. In Section 3, we prove the RN theorem for regular stable sets $\mathcal{S}$ in an SDRS $\mathcal{R}$, and demonstrate that if $\mathcal{R}$ is not stable, then the theorem fails. In Section 4, we introduce DFSs, compare them with SDRSs, and prove the Relative Normalization and Optimality theorems for any stable set $\mathcal{S}$ of results. Section 5 gives a characterization of zig-zag relation via extraction in non-duplicating SDRSs, used in Section 6 to prove that, in every non-duplicating SDRS, zig-zag is a family relation, and moreover, is the unique separable family-relation. In Section 7 we define and study implementation DRSs, and show the (de)compositionality of sharing. Conclusions appear in Section 8.

It should be remarked that it is not the aim of this paper to provide a comprehensive abstract theory of normalization and optimality that can readily be applied to particular (syntactic) systems covered by SDRSs and DFSs. In particular, it cannot replace the syntactic studies of normalization and optimality in the literature (part of which was mentioned above): some of our axioms are non-trivial to verify for particular systems, and we do not cover all aspects of the subject. Rather, we try to bring out and analyze some of the main principles and methods on which the theory of normalization and optimality is based. And although our abstract framework is very general, our results and proofs are far from trivial and do save considerable work for particular systems.

The results of this paper have been reported in [20,21,41,45]. Some results are taken from [39,40]

2 Deterministic Residual Structures

In this section we define *Deterministic Residual Structures* (DRSs) which are *Abstract Reduction Systems* (ARSs) satisfying certain properties concerning *residuals*. Residuals of redexes were first introduced and studied by Church and Rosser in the λI-calculus [14].

The definition and some results about ARSs can be found e.g., in [47,28]. Our definition is slightly different, and follows that of Hindley [26].

Definition 1 *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 $\rightarrow$: $Red \rightarrow (Ter \times Ter)$ is a function such that for any $t \in Ter$ there is only a finite set of $u \in Red$ such that $\rightarrow (u) = (t, s)$, 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 [47] 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 $t \stackrel{P}{\twoheadrightarrow} s$ if P denotes a reduction (sequence) from t to s. $|P|$ denotes the length of P. $P + Q$ denotes composition (or concatenation) of P and Q. We use U, V, W to denote sets of redexes of a term.*

DRSs are similar to *Determinate Concurrent Transition Systems* (DTCS) of Stark [68], to *Abstract Reduction Systems* (ARSs) of Gonthier et al. [24], and to van Oostrom's *orthogonal Descendant Rewriting Systems* [61]. The main difference from DCTSs is that Stark requires a non-duplicating residual relation, and distinguishes a start state. Unlike ARSs of [24], we do not have a nesting relation on redexes and the corresponding axioms, and the stability axiom is modified accordingly. Instead, we study properties of conflict-free transition and reduction systems based on the duplication and erasure effect of executed transitions on the others, and develop a theory that is applicable to systems with nested transitions too. The difference from van Oostrom's orthogonal DRSs is that in the latter the notion of *descendant* of any subterm/position of a term is formalized, not only the notion of residual of redexes; and the [weak acyclicity] axiom below is not required there. Closely related abstract reduction systems are studied in [54,65].

Definition 2 (Deterministic Residual Structure) *A* Deterministic Residual Structure *(DRS) is a pair $\mathcal{R} = (A, /)$, 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, *and if moreover $v \neq u$, then u* discards v. *A redex of s which is not a residual of any $v \in t$ under u is said to be u-*new *or* created *by u. The set v/P of residuals of v under any reduction P is defined by transitivity.*

A development *of $U \subseteq t$ is a reduction $P : t \twoheadrightarrow$ that only contracts residuals of redexes from U; it is* complete *if $U/P = \cup_{u \in U} u/P = \emptyset$. 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:*

[FD] *(*Finite Developments *[24]) All developments are terminating; all complete developments of $U \subseteq t$ end at the same term; and residuals of a redex $v \in t$ under all complete developments of U are the same.*
[weak acyclicity] *([68]) Let $u, v \in t$, $u \neq v$, and $u/v = \emptyset$. Then $v/u \neq \emptyset$.*[1]

We call a DRS $\mathcal{R}$ stable *(SDRS) if the following axiom is satisfied:*

[stability] *If $u, v \in t$ are different redexes, $t \xrightarrow{u} e$, $t \xrightarrow{v} s$, and u creates a redex $w \in e$, then the redexes in $w/(v/u)$ are not u/v-residuals of redexes of s, i.e., they are created along u/v.*

$$\begin{tikzcd} t \arrow[r, "u"] \arrow[d, "v"'] & e \arrow[r, "w"] \arrow[d, two heads, "v/u"] & \cdot \\ s \arrow[r, two heads, "u/v"'] & o \arrow[r, two heads, "w/(v/u)"'] & \cdot \end{tikzcd}$$

One can verify that all orthogonal (first or higher-order, see e.g., [65]) term rewriting systems, such as the λ-calculus, form DRSs. These systems are stable, and can be shown so just using an appropriate notion of descendant which assigns the contractum to the contracted redex – labelling is not necessary. Further, orthogonal Term Graph Rewriting Systems [33], which are equivalent to orthogonal DAG (Directed Acyclic Graph) rewriting systems defined

[1] This axiom is called [acyclicity] in [20], and is axiom (4) in [68].

via labelled orthogonal TRSs [52,53], are DRSs but they do not satisfy all the nesting axioms of [24]. Similarly, Interaction Nets [48] form DRSs (when restricted to finite nets), and they do not have or need any nesting order on redexes (as nets are undirected, and possibly cyclic, graphs).

The properties of the residual relation in orthogonal systems are all standard [29,50,12,51,13,68,54], and we only review quickly the main constructions used in this paper. In a DRS $\mathcal{R}$, *Lévy-equivalence* or *permutation-equivalence* is defined as the smallest relation on co-initial finite 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 sets of redexes in the same term. The residual relation on redexes is extended to all co-initial reductions as follows: $(P_1+P_2)/Q = P_1/Q+P_2/(Q/P_1)$ and $Q/(P_1+P_2) = (Q/P_1)/P_2$. Further, the *Lévy-embedding* relation on co-initial finite reductions, $\trianglelefteq_L$, is defined by: $P \trianglelefteq_L Q$ iff $P/Q = \emptyset$. One can show that $P \approx_L Q$ iff $P \trianglelefteq_L Q$ and $Q \trianglelefteq_L P$ and that $P \trianglelefteq_L Q$ iff $Q \approx_L P+N$ for some N. Intuitively, $P \trianglelefteq_L Q$ expresses that Q does more work than P, and Q/P is the part of Q that remains after P has been done. Finally, one can prove the following *strong Church-Rosser* theorem for DRSs: For all co-initial finite reductions P and Q in a DRS, $P \sqcup Q \approx_L Q \sqcup P$, where $P \sqcup Q$ abbreviates $P + Q/P$.

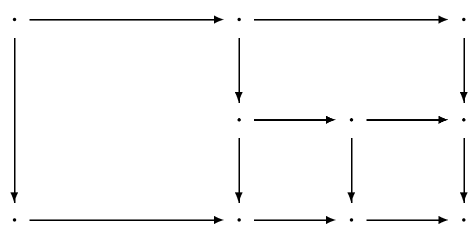
To show that the above definitions are sound, it is convenient to consider *multi-step* reductions, where a multi-step contracts *simultaneously*, or in *parallel*, a set of redexes in a term; thus a multi-step can be seen as a complete development of a set of redexes, and it is conventional to switch freely between multi-steps and complete developments. Now for example correctness of the above definition of residuals for reductions can be proved by induction on the number of multi-steps in P_1, P_2 and Q, where steps in these reductions are viewed as multi-steps contracted singleton sets of redexes. These induction is equivalent to induction on *multi-diagrams*, which are diagrams of the form:

$$\begin{tikzcd}
\cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[d, two heads] \\
\cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[d, two heads] \\
\cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[d, two heads] \\
\cdot \arrow[r, two heads] & \cdot \arrow[r, two heads] & \cdot \arrow[r, two heads] & \cdot \arrow[r, two heads] & \cdot
\end{tikzcd}$$

where each arrow represents a multistep, and each inner square is yielded by the equation $U+V/U \approx_H V+U/V$, where U and V as well as U/V and V/U are multi-steps. Here $\approx_H$ is *Hindley-equivalence*, which is defined for finite co-initial reductions as follows: $P \approx_H P'$ iff they end at the same term and for any redex u in the initial term, the residuals of u under P and under P' are the same redexes. The induction principle is then to infer a property for such a multi-diagram from properties of its proper sub-diagrams. For example, the correctness of definitions $(P_1+P_2)/Q = P_1/Q+P_2/(Q/P_1)$ and $Q/(P_1+P_2) = (Q/P_1)/P_2$ can be inferred from the following multi-diagram:

$$\begin{tikzcd}
\cdot \arrow[r, ->>, "P_1"] \arrow[d, ->>, "Q"'] & \cdot \arrow[r, ->>, "P_2"] \arrow[d, ->>, "Q/P_1"] & \cdot \arrow[d, ->>, "Q/(P_1+P_2)"] \\
\cdot \arrow[r, ->>, "P_1/Q"'] & \cdot \arrow[r, ->>, "P_2/(Q/P_1)"'] & \cdot
\end{tikzcd}$$

However, it is not necessary to switch to multi-steps in order to provide a sound induction principle for definitions and proofs involving many-step reductions. (Otherwise, the 'trick' of using multi-step reductions would not be sound!) Klop's *commutative diagrams* can be used instead [46]. Klop diagrams are constructed using developments rather than multi-steps, and this involves 'splitting' of elementary (or one-step) diagrams. The following is a simple example of a Klop-diagram:



where arrows now represent reduction steps, not multi-steps, and each inner square is again yielded by the equation $U+V/U \approx_H V+U/V$ where U and V as well as U/V and V/U are now complete developments of respective sets of redexes. Both multi-step and Klop diagrams use empty (multi-) steps to retain the rectangular shape of the diagrams. Construction of a Klop-diagram need not terminate in general, but it always does for reductions in DRSs, by virtue of the [FD] axiom. Construction of Klop diagrams is also well described in [8], for the case of the λ-calculus. We will often use Klop diagrams in proofs. In which case P/Q, say is considered as a reduction rather than a multi-step reduction, and is thus only unique up to the particular sequentializations of corresponding multi-steps. (However, P/Q is defined more precisely than up to $\approx_L$.) In conclusion, we note that instead of [FD] in the definition of DRSs, we could require equivalently only termination of developments and the following strong local commutativity property: for all co-initial redexes u and v, $u \sqcup v \approx_H v \sqcup u$.

The [stability] axiom is not used in the above constructions. Intuitively, stability means that a redex cannot arise from two unrelated sources. This property has a natural extension to many-step reductions, where 'unrelated' is formalized by the notion of *external* which captures the concept that two external reductions do not contract redexes that have common residuals (although the contracted redexes may have 'inessential' common ancestors).

Definition 3 • *Let $u \in U \subseteq t$ and $P : t \twoheadrightarrow o$. We call P* external *to U (resp. u) if P does not contract residuals of redexes in U (resp. residuals of u).*

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

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

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 $P : t \overset{a}{\rightarrow} f(g(a')) \overset{f}{\rightarrow} b$, $Q : t \overset{a}{\rightarrow} f(g(a')) \overset{g}{\rightarrow} f(c)$, and $N : t \overset{g}{\rightarrow} f(c)$. 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 4 *Let $P : t_0 \overset{v_0}{\rightarrow} t_1 \overset{v_1}{\rightarrow} \ldots \to t_n$ be external to $U = \{u_1, \ldots, u_n\} \subseteq t_0$, and let $Q_0 : t_0 \twoheadrightarrow o_0$. Then $P' = P/Q_0$ is external to the set $U' = U/Q_0$.*

Proof *Let $P_i : t_0 \overset{v_0}{\rightarrow} t_1 \overset{v_1}{\rightarrow} \ldots \to t_i$, $Q_i = Q_0/P_i$, and $P'_{i+1} = v_i/Q_i$,* $(0 \le i < n)$ *(see the figure). Since P is external to U, we have for each i that $v_i \notin U/P_i$. Therefore, $v_i/Q_i \cap U/(P_i + Q_i) = \emptyset$ (since the residuals of different redexes are different). Thus $v_i/Q_i \cap U/(Q_0 + P'_1 + \ldots + P'_i) = \emptyset$. Hence, P'_{i+1} is external to $U'/(P'_1 + \ldots + P'_i)$. This means that P' is external to U'.*

$$
\begin{tikzcd}
t_0 \arrow[r, "v_0 = P_1"] \arrow[d, "Q_0"'] & t_1 \arrow[r, two heads] \arrow[d, "Q_1"'] & t_{n-1} \arrow[r, "v_{n-1}"] \arrow[d, "Q_{n-1}"'] & t_n \arrow[d, "Q_n"] \\
o_0 \arrow[r, two heads, "P'_1"'] & o_1 \arrow[r, two heads] & o_{n-1} \arrow[r, two heads, "P'_n"'] & o_n
\end{tikzcd}
$$

We conclude this section by proving two fundamental lemmas, which generalize the [weak acyclicity] and [stability] axioms to many-step reductions, and are used repeatedly throughout the paper

Lemma 5 (Weak Acyclicity) *Let P, N be co-initial finite reductions in a DRS, and let P be external to N. Then $N \not\lesssim_L P$.*

Proof *We show that $P/N = \emptyset$ implies $N/P \neq \emptyset$ by induction on the number of elementary diagrams in a Klop diagram $D(P, N)$ of P and N. It follows from Definition 3 that the top and left edges of all sub-diagrams of $D(P, N)$ are external reductions, therefore the induction is sound. So let $P/N = \emptyset$. If $|P| = |N| = 1$, we have $N/P \neq \emptyset$ by [weak acyclicity]. Otherwise, let say $|P| > 1$, let $P : t \overset{Q}{\twoheadrightarrow} e \overset{v}{\rightarrow} o$, and let $N^* = N/Q$ (see the figure). By the induction assumption, $N^* \neq \emptyset$. So let $N^* = N' + w$, where w is the last (non-empty) step of N^*, and let $V = v/N'$. Again by the induction assumption, $w/V \neq \emptyset$. Hence $N/P \neq \emptyset$.*

$$\begin{tikzcd}
t \arrow[r, two heads, "Q"] \arrow[dd, two heads, "N"'] & e \arrow[r, "v"] \arrow[d, two heads, "N'"'] & o \arrow[d, two heads] \\
 & \cdot \arrow[r, two heads, "V"] \arrow[d, "w"'] & \cdot \arrow[d, two heads] \\
s \arrow[r, two heads, "\emptyset"'] & s \arrow[r, two heads, "\emptyset"'] & s
\end{tikzcd}$$

Lemma 6 (Stability) *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.*

$$\begin{tikzcd}
t \arrow[r, two heads, "P"] \arrow[d, two heads, "Q"'] & s \supseteq W \arrow[d, two heads, "Q/P"] \\
e \arrow[r, two heads, "P/Q"'] & o \supseteq W/(Q/P)
\end{tikzcd}$$

Proof *By induction on the number n of elementary diagrams in the Klop diagram $D(P, Q)$ of P and Q. The case $n = 1$ is the axiom [stability]. It follows from Definition 3 that the top and left edges of all sub-diagrams of $D(P, Q)$ are external reductions, therefore we can assume the lemma is proved for all subdiagrams of $D(P, Q)$. Let $W^* = W/(Q/P) \neq \emptyset$.*

If $|P| = 1$ and $|Q| > 1$, say $Q = v + Q'$, then $W' = W/(v/P) \neq \emptyset$, and by [stability] redexes in W' are created by P/v. By the induction assumption, redexes in W^* are created by P/Q and $Q'/(P/v)$ is external to W', implying *that $Q/P = v/P + Q'/(P/v)$ is external to W.*

$$
\begin{tikzcd}
t \arrow[r, "v"] \arrow[d, "P"'] & \cdot \arrow[r, two heads, "Q'"] \arrow[d, "P/v"] & e \arrow[d, two heads, "P/Q"] \\
W \subseteq s \arrow[r, two heads, "v/P"'] & s' \ni W' \arrow[r, two heads, "Q'/(P/v)"'] & \cdot
\end{tikzcd}
$$

Now it remains to consider the case when $|P| > 1$, i.e., $P = u + N$ with $|N| > 0$ (see the figure below). Then $W = W_u/N \cup W_N$, where W_u is the set of redexes created by u, and W_N is the set of redexes created by (i.e., along) N. By the induction assumption, Q/u is external to W_u, and the redexes in $W_u/(Q/u)$ are created by u/Q. By Lemma 4, $Q/P = (Q/u)/N$ is external to the set W_u/N. By the induction assumption, $Q/P = (Q/u)/N$ is external to W_N and redexes in $W_N/(Q/P)$ are created by $N/(Q/u)$. Hence Q/P is external to W, and since $W/(Q/P) = W_N/(Q/P) \cup W_u/((Q/u) \sqcup N)$, redexes in $W/(Q/P)$ are created by P/Q.

$$
\begin{tikzcd}
t \arrow[r, two heads, "Q"] \arrow[d, "u"'] & e \arrow[d, "u/Q"] \\
W_u \in o \arrow[r, two heads, "Q/u"] \arrow[d, two heads, "N"'] & o' \ni W_u/(Q/u) \arrow[d, two heads, "N/(Q/u)"] \\
W \in s \arrow[r, two heads, "(Q/u)/N"'] & s' \ni W^*
\end{tikzcd}
$$

3 Relative Normalization for regular stable sets

In this section, we prove that, for any *regular* stable set of terms $\mathcal{S}$ in a stable DRS $\mathcal{R}$, an $\mathcal{S}$-normal form of an $\mathcal{S}$-normalizable term can be found by contracting $\mathcal{S}$-needed redexes in it, even if every $\mathcal{S}$-needed step is preceded by a finite number of $\mathcal{S}$-unneeded steps. We show that without the assumption of stability for $\mathcal{R}$, this result breaks down.

Definition 7 Let $\mathcal{S}$ be a set of terms in a DRS $\mathcal{R}$.

(1) We call a redex $u \in t$ *$\mathcal{S}$-needed* if at least one residual of it is contracted in any reduction from t to a term in $\mathcal{S}$, and call u *$\mathcal{S}$-unneeded* otherwise.
(2) We call $\mathcal{S}$ *stable* if:

(a) $\mathcal{S}$ is *closed under parallel moves*:
for any $t \notin \mathcal{S}$, any $P : t \twoheadrightarrow o \in \mathcal{S}$, and any $Q : t \twoheadrightarrow e$ which does not contain terms in $\mathcal{S}$, the final term of P/Q is in $\mathcal{S}$; and

(b) $\mathcal{S}$ is *closed under unneeded expansion*:
for any $e \xrightarrow{u} o$ such that $e \notin \mathcal{S}$ and $o \in \mathcal{S}$, u is $\mathcal{S}$-needed.

(3) We call a stable set $\mathcal{S}$ *regular* if in no term can an $\mathcal{S}$-unneeded redex duplicate an $\mathcal{S}$-needed one.

A stable set need not be closed under reduction – Q/P in the definition above may contain terms not in $\mathcal{S}$, but closure under parallel moves requires that the final term is. Stability and regularity coincide in non-duplicating systems. Below $\mathcal{S}$ will usually denote a stable set of terms in some DRS. For simplicity, we only consider stable sets that are closed under reduction, although the theorem holds for all stable sets as shown in [18,22]; obviously, closure under reduction implies closure under parallel moves.

Examples of stable sets are normal forms [29], head-normal forms in the λ-calculus [10], weak-head-normal forms in the λ-calculus, constructor-head-normal forms for constructor TRSs [59], and root-stable forms in orthogonal TRSs [57]. All the above sets are closed under reduction, and are regular. Other examples include weak-head-normal forms (up to garbage-collection, modulo a congruence) in Yoshida's λf-calculus (an environment calculus) [73], the set of *answers* in the *call-by-need* λ-calculus of Ariola et al. [3], and flexible generalized-head-normal forms in the λ^v_{hd}-calculus of Xi [72]; all are conditional rewrite systems. An example of an orthogonal TRS with an irregular stable $\mathcal{S}$ is given in Remark 16.

We begin the proof by showing that $\mathcal{S}$-unneeded redexes cannot create $\mathcal{S}$-needed ones, and that residuals of $\mathcal{S}$-unneeded redexes remain unneeded. When $\mathcal{S}$ is regular, this enables us to construct an $\mathcal{S}$-needed variant of any $\mathcal{S}$-normalizing reduction.

Lemma 8 *For any stable $\mathcal{S}$, residuals of $\mathcal{S}$-unneeded redexes of a term t under any reduction $P : t \twoheadrightarrow s$ remain $\mathcal{S}$-unneeded.*

Proof *Let $u \in t$ be $\mathcal{S}$-unneeded, and let v be a P-residual of u. Then there is an $\mathcal{S}$-normalizing reduction $Q : t \twoheadrightarrow o$ that is external to u. By Lemma 4, Q/P is external to v, thus v is $\mathcal{S}$-unneeded (as Q/P is $\mathcal{S}$-normalizing by stability of $\mathcal{S}$).*

Lemma 9 Let $\mathcal{S}$ be stable, let $t \notin \mathcal{S}$, let $t \xrightarrow{u} e$, let u be $\mathcal{S}$-unneeded, and let $w \in e$ be a redex created by u, in a stable DRS. Then w is $\mathcal{S}$-unneeded.

Proof If $e \in \mathcal{S}$, then every redex in e, including w, is $\mathcal{S}$-unneeded. But suppose $e \notin \mathcal{S}$: Since u is $\mathcal{S}$-unneeded, there is an $\mathcal{S}$-normalizing $P : t \twoheadrightarrow s$ that does not contract residuals of u. By Lemma 6, P/u does not contract residuals of

w. Also, P/u is $\mathcal{S}$-normalizing since $\mathcal{S}$ is closed under parallel moves. Hence w is $\mathcal{S}$-unneeded.

The following example shows that, in the above lemma, stability of the DRS is necessary.

Example 10 *Let terms in the DRS $\mathcal{R}$ be $t = I(I(x))$, $s = I(x)$, and $e = x$; redexes in t be $u = t$ and $v = I(x)$, s contain the only redex $w = s$, and x does not contain a redex; let the reduction relation be given by $Red = \{t \xrightarrow{u} s,\ t \xrightarrow{v} s,\ s \xrightarrow{w} x\}$, let the residual relation be empty except for empty reductions, for which the residual relation is identity, and let $\mathcal{S} = \{x\}$. (Obviously, this is not, and cannot be, the usual residual relation for orthogonal TRSs.) Then $\mathcal{S}$ is stable and regular, both u and v are $\mathcal{S}$-unneeded, and both create the redex $w \in s$ that is $\mathcal{S}$-needed. Note also that the Relative Hypernormalization theorem (proved below) is not valid for $(\mathcal{R}, \mathcal{S})$ since $t \notin \mathcal{S}$ is $\mathcal{S}$-normalizable but does not contain an $\mathcal{S}$-needed redex.*

Definition 11 We call $P : t_0 \to t_1 \to \ldots$ $\mathcal{S}$-*(un)needed* if it contracts only $\mathcal{S}$-(un)needed redexes. We call P $\mathcal{S}$-*quasi-needed* if it contracts infinitely many $\mathcal{S}$-needed redexes, and call it $\mathcal{S}$-*semi-needed* if it can be expressed as $P = P_1 + P_2$ where P_1 is $\mathcal{S}$-needed and P_2 is $\mathcal{S}$-unneeded. In the latter case, we call P_1 the $\mathcal{S}$-*needed part* of P (P_1 can be infinite, in which case $P_2 = \emptyset$).

We now describe an algorithm that, for a *regular* stable set of terms $\mathcal{S}$ in an SDRS $\mathcal{R}$, transforms any finite or infinite reduction P into an $\mathcal{S}$-semi-needed reduction $K(P)$. The algorithm is as follows: find in P the leftmost subreduction $P_0 : t \xrightarrow{u} s \xrightarrow{v} o$ such that u is $\mathcal{S}$-unneeded and v is $\mathcal{S}$-needed. Let $P = P_1 + P_0 + P_2$. By Lemma 9, v is a residual of a redex $v' \in t$, which is $\mathcal{S}$-needed by Lemma 8. Since $\mathcal{S}$ is regular, v is the only residual of v', hence P_0 and $P_0' = v' + u/v'$ are both complete developments of the set $u, v' \in t$, thus $P_0 \approx_L P_0'$. Now replace P_0 by P_0' in P. Transform the obtained reduction P' in the same way, and so on as long as possible. Obviously, by regularity of $\mathcal{S}$, the number of $\mathcal{S}$-unneeded steps in P' preceding v' is less than the number preceding v in P, and the number of $\mathcal{S}$-needed steps in P and P' coincide.

For the above transformation procedure to terminate, instead of regularity of $\mathcal{S}$ it would be enough to require that an $\mathcal{S}$-needed redex had at most one $\mathcal{S}$-needed residual under an $\mathcal{S}$-unneeded step. Termination of the transformation procedure is crucial for our method for proving termination of any $\mathcal{S}$-needed reduction starting from an $\mathcal{S}$-normalizable term.

Lemma 12 Let P be a finite or infinite reduction in an SDRS, and let $\mathcal{S}$ be regular.

(1) If P ends at a term in $\mathcal{S}$, then $K(P)$ is a finite $\mathcal{S}$-semi-needed reduction whose $\mathcal{S}$-needed part ends at a term in $\mathcal{S}$ as well.

(2) If P is $\mathcal{S}$-quasi-needed, then $K(P)$ is an infinite $\mathcal{S}$-needed reduction.

Proof

(1) Since the transformation K does not change the number of $\mathcal{S}$-needed steps in P, it follows that $K(P)$ is $\mathcal{S}$-semi-needed, and it ends at $\mathcal{S}$ since $K(P) \approx_L P$. The step of $K(P)$ entering $\mathcal{S}$ is the last $\mathcal{S}$-needed step of $K(P)$ by stability of $\mathcal{S}$.

(2) Immediate from the construction of $K(P)$.

Next we show that, unless it is contracted, an $\mathcal{S}$-needed redexes has at least one $\mathcal{S}$-needed residual. Therefore, residuals of $\mathcal{S}$-quasi-needed reductions remain so. It follows that an $\mathcal{S}$-normalizable term cannot possess an $\mathcal{S}$-quasi-needed reduction.

Lemma 13 Let $\mathcal{S}$ be a regular stable set of terms in a DRS $\mathcal{R}$, and let $t \xrightarrow{u} s$. Then any $\mathcal{S}$-needed redex $v \in t$ different from u has an $\mathcal{S}$-needed residual.

Proof If t is not $\mathcal{S}$-normalizable, then neither is s, and all redexes in t and s are $\mathcal{S}$-needed. So suppose t is $\mathcal{S}$-normalizable ($t \notin \mathcal{S}$ since t contains an $\mathcal{S}$-needed redex), and suppose on the contrary that each residual v_i of v in s is $\mathcal{S}$-unneeded. By closure of $\mathcal{S}$ under parallel moves, s is $\mathcal{S}$-normalizing too. By Lemma 12.(1), there is an $\mathcal{S}$-needed $\mathcal{S}$-normalizing reduction $P : s \twoheadrightarrow o$. Since by Lemma 8 all residuals of each v_i along P are $\mathcal{S}$-unneeded, P is external to all v_i. Therefore, $u + P$ is external to v and is $\mathcal{S}$-normalizing – a contradiction, since v is $\mathcal{S}$-needed.

Lemma 14 Let t_0 have an $\mathcal{S}$-quasi-needed reduction and $t_0 \xrightarrow{u} s_0$. Then s_0 also has an $\mathcal{S}$-quasi-needed reduction.

Proof By Lemma 12, t_0 has an infinite $\mathcal{S}$-needed reduction $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots$. Let $U_i = u/(u_0 + \ldots + u_{i-1})$, $i = 0, 1, \ldots$ (see the diagram below). It follows from finiteness of developments that there are infinitely many numbers k such that $u_k \notin U_k$ (otherwise there should be a number m such that $t_m \xrightarrow{u_m} t_{m+1} \xrightarrow{u_{m+1}} \ldots$ is an infinite U_m-development). By Lemma 13, u_k has at least one $\mathcal{S}$-needed U_k-residual in s_k, i.e. u_k/U_k contains at least one $\mathcal{S}$-needed step. Hence P/u is $\mathcal{S}$-quasi-needed.

$$
\begin{tikzcd}
t_0 \arrow[r, "u_0"] \arrow[d, two heads, "u = U_0"'] & t_1 \arrow[r, "u_1"] \arrow[d, two heads, "U_1"'] & t_2 \arrow[r, two heads] \arrow[d, two heads, "U_2"'] & \cdot \\
s_0 \arrow[r, two heads, "u_0/U_0"'] & s_1 \arrow[r, two heads, "u_1/U_1"'] & s_2 \arrow[r, two heads] & \cdot
\end{tikzcd}
$$

Theorem 15 (Relative Hypernormalization) Let $\mathcal{S}$ be a regular stable set of terms in a stable DRS $\mathcal{R}$, and let $t \notin \mathcal{S}$ be a term in $\mathcal{R}$. Then

(1) t contains at least one $\mathcal{S}$-needed redex.
(2) t has an $\mathcal{S}$-normal form iff it does not possess a reduction in which infinitely many times $\mathcal{S}$-needed redexes are contracted.

Proof

(1) By Definition 7 if t is not $\mathcal{S}$-normalizing, and by Lemma 12 otherwise.
(2) ($\Rightarrow$) Let $t \overset{P}{\twoheadrightarrow} s \in \mathcal{S}$. Suppose on the contrary that there is an $\mathcal{S}$-quasi-needed Q starting from t. Then by Lemma 14 Q/P is $\mathcal{S}$-quasi-needed as well – a contradiction, since all terms of Q/P are in $\mathcal{S}$, by the closure of $\mathcal{S}$ under reduction, and therefore Q/P must be $\mathcal{S}$-unneeded.
($\Leftarrow$) By (1), one can repeatedly contract $\mathcal{S}$-needed redexes in t, unless a term in $\mathcal{S}$ is reached; the latter is inevitable since t does not have an infinite $\mathcal{S}$-needed reduction.

Remark 16 If $\mathcal{S}$ is not regular, then Lemma 12 need not hold. Indeed, consider the example from [18,22]: take $R = \{f(x) \to h(f(x), f(x)),\ a \to b\}$ and take for $\mathcal{S}$ the set of terms not containing occurrences of a. It is easy to check that $\mathcal{S}$ is stable, but is not regular, since the outermost redex in $t = f(a)$ is $\mathcal{S}$-unneeded, while the innermost one is $\mathcal{S}$-needed. Then the reduction $P : f(a) \to h(f(a), f(a)) \to h(f(b), f(a)) \to h(f(b), h(f(a), f(a))) \to h(f(b), h(f(b), f(a))) \to \ldots$ is $\mathcal{S}$-quasi-needed, while the $\mathcal{S}$-needed part $Q : f(a) \to f(b)$ of $K(P)$ is $\mathcal{S}$-normalizing, and $P/Q = f(b) \to h(f(b), f(b)) \to h(f(b), h(f(b), f(b))) \to \ldots$ is $\mathcal{S}$-unneeded, thus not $\mathcal{S}$-quasi-needed any more. Because of that, the proof of Lemma 14 fails, and the $\mathcal{S}$-needed strategy need not be hypernormalizing.

4 Relative Normalization and Optimality in Deterministic Family Structures

In order to generalize the RN theorem to all stable sets in DRSs, and to prove a Relative Optimality theorem, we introduce *Deterministic Family structures* (DFSs) by defining a notion of *family* in a DRS, and by imposing some axioms on the *contribution* relation on families. This enables us to repeat the proof of the RN theorem in [18,22] for all DFSs, and makes explicit the properties of a family relation needed to develop an abstract theory of optimal normalization.

Definition 17 (Deterministic Family Structure) *A* Deterministic Family Structure *(DFS) is a triple $\mathcal{F} = (\mathcal{R}, \simeq, \hookrightarrow)$, where $\mathcal{R}$ is a DRS; the family relation $\simeq$ is an equivalence relation on redexes with* histories*; and $\hookrightarrow$ is the* contribution *relation on co-initial families, defined as follows:*

(1) *For any co-initial reductions P and Q, a redex Qv in the final term of Q*

(read as v with history Q*) is called a* copy *of a redex Pu if $P \trianglelefteq_L Q$, i.e., $P+Q/P \approx_L Q$, and v is a Q/P-residual of u; the* zig-zag *relation $\simeq_z$ is the symmetric and transitive closure of the copy relation. The* family *relation $\simeq$ is an equivalence relation among redexes with histories containing $\simeq_z$. A* family *is an equivalence class of the family relation; families are ranged over by $\phi, \psi, \ldots$. $Fam(\ )$ denotes the family of its argument.*

(2) *Further, $\simeq$ and $\hookrightarrow$ satisfy the following axioms:*

[initial] *Let $u, v \in t$ and $u \neq v$, in $\mathcal{R}$. Then $Fam(\emptyset_t u) \neq Fam(\emptyset_t v)$, where $\emptyset_t$ is the empty reduction starting from t.*

[contribution] *$\phi \hookrightarrow \phi'$ iff for any $Pu \in \phi'$, P contracts at least one redex in ϕ.*

[creation] *Let $e \overset{P}{\twoheadrightarrow} t \overset{u}{\rightarrow} s$, and let u create $v \in s$. Then $Fam(u) \hookrightarrow Fam(v)$, or more precisely, $Fam(Pu) \hookrightarrow Fam((P+u)v)$.*

[FFD] *(*Finite Family Developments*) Any reduction that contracts redexes of a finite number of families is terminating.*[2]

Note that the [contribution] can be viewed as a definition of $\hookrightarrow$ rather than as an axiom. Hence sometimes we will consider a DFS as a pair $\mathcal{F} = (\mathcal{R}, \simeq)$.

One can check that all the existing definitions of family relation in the literature [50,46,32,53,4,62] satisfy the above axioms, thus our definition is consistent. Indeed, validity of the first three axioms follow immediately from the structure of labels in the labelling definitions of these families. The axiom [FFD] had been proved separately for all the above systems using the syntax in [50,46,63]. An axiomatic approach to [FFD] is developed in [54], but it is not fully general as for example it does not apply to all orthogonal higher-order or graph rewriting systems [63,54]. Since there are many different (and perhaps incompatible) ways to construct a DFS from a stable DRS, and since we want to be as general as (reasonably) possible, we do not investigate here any finer axiomatization of the residual relation that would imply [FFD]. The reason for considering notions of family other than just the zig-zag is that we want to be more flexible and able to consider a large class of sharing mechanisms as legal. Further, there are well-behaved sharing mechanisms, such as the one in [4], that are strictly larger than zig-zag (when we regard the reduction system as a DRS). We will comment on this below. Moreover, as we will see, our sharing concept has the nice property of compositionality, facilitating study of complex concepts of family.

Despite its generality, our family concept does not cover all possible sharing concepts studied in the literature. For example, consider the TRS from [27] for computing Fibonacci numbers, having the following three rules: $fib(0) \rightarrow 0$, $fib(succ(0)) \rightarrow succ(0)$ and $fib(succ(succ(n))) \rightarrow fib(succ(n)) + fib(n)$, and rules for functions $0, succ$ and $+$; and consider the reduction:

[2] This axiom is called [termination] in [20].

$$\begin{array}{ll} & fib(succ^4(0)) \\ \to & fib(succ^3(0)) + fib(succ^2(0)) \\ \to & fib(succ^2(0)) + fib(succ^1(0)) + fib(succ^2(0)) \\ \to & \ldots \end{array}$$

In the Jungle Graph implementation of that TRS [27] (which allows 'accidental' sharing of possibly completely unrelated, but graphically equal, subterms), it is possible to share the two occurrences of $fib(succ^2(0))$ in the last term, but (by [contribution]) the two occurrences cannot be considered as members of the same family since (by [creation]) they have different contributor families.

Let us call $\lceil\phi\rceil^{\leq} = \{\phi_i \mid \phi_i \hookrightarrow \phi\}$ the *cone* of ϕ. It follows immediately from the family axioms that:

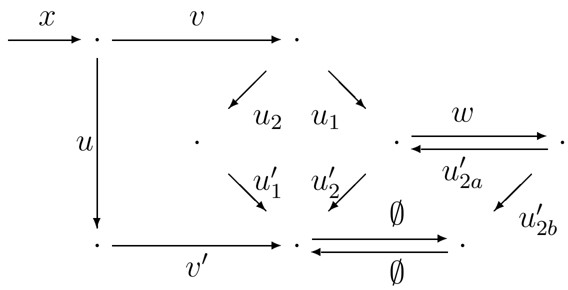
Proposition 18 *In any DFS $\mathcal{F}$:*

[irreflexivity] $\phi \not\hookrightarrow \phi$.
[transitivity] *If $\phi \hookrightarrow \phi'$ and $\phi' \hookrightarrow \phi''$, then $\phi \hookrightarrow \phi''$.*
[cone] $\phi \hookrightarrow \psi \Rightarrow \lceil\phi\rceil^{\leq} \subset \lceil\psi\rceil^{\leq}$. *(i.e., $\lceil\phi\rceil^{\leq} \subseteq \lceil\psi\rceil^{\leq}$ and $\lceil\phi\rceil^{\leq} \neq \lceil\psi\rceil^{\leq}$).*
[finiteness] *For any ϕ, $\lceil\phi\rceil^{\leq}$ is finite, and $\lceil\emptyset_t u\rceil^{\leq} = \emptyset$ for any $u \in t$.*
[antisymmetry] *If $\lceil\phi\rceil^{\leq} = \lceil\psi\rceil^{\leq}$ and $\phi \neq \psi$, then $\phi \not\hookrightarrow \psi$ (and $\psi \not\hookrightarrow \phi$).*

Proof *[irreflexivity], [transitivity], and [finiteness] follow immediately from [contribution]. [cone] follows from [irreflexivity] and [transitivity]. [antisymmetry] follows from [cone].*

The following example shows that, in a stable DRS with $\simeq$ and $\hookrightarrow$, [initial], [creation] and [contribution] do not imply [FFD].

Example 19 Consider the ARS $\mathcal{R}$ given by the figure below, where the redex x creates u and v; u_1 and u'_{2a} create w; $v/u = v'$; $u/v = \{u_1, u_2\}$; $u_1/u_2 = u'_1$, $u_2/u_1 = u'_2$; $u'_2/w = \{u'_{2a}, u'_{2b}\}$; $w/u'_2 = \emptyset$, $u'_{2a}/u'_{2b} = \emptyset$, $u'_{2b}/u'_{2a} = u'_2$. All the us are residuals of u, and hence belong to the same family ϕ_u. Similarly, v and v' must be in the same family too, say ϕ_v. Further, take $\phi_x = \{x\}$, take for ϕ_w the set of all contracted ws (with histories), and define the contribution relation on $\phi_x, \phi_u, \phi_v, \phi_w$ by $\phi_x \hookrightarrow \phi_v, \phi_u$ and $\phi_u \hookrightarrow \phi_w$. Since the only infinite reduction goes through the cycle infinitely many times, and each time the contracted w is *created* by u'_{2a}, all developments in the figure are terminating. It remains to note that [FD] and the other family axioms except [FFD] are satisfied too. Note that the DRS $\mathcal{R}$ is stable.

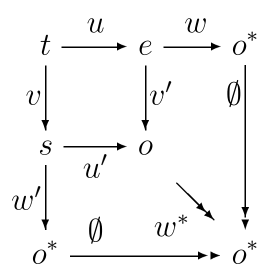


Lemma 20 *Any DFS $\mathcal{F}$ is a stable DRS.*

Proof *We want to show that if $u, v \in t$ are different redexes, $t \xrightarrow{u} e$, $t \xrightarrow{v} s$, $e \overset{v/u}{\twoheadrightarrow} o$, and u creates a redex $w \in e$, then the redexes in $w/(v/u) \in o$ are not u/v-residuals of redexes of s. By axioms [creation] and [contribution], for any redex $w' \in s$, $\lceil Fam(w') \rceil^{\leq} = \emptyset$ if w' is not a created redex, and $\lceil Fam(w') \rceil^{\leq} = \{Fam(v)\}$ otherwise; and $\lceil Fam(w) \rceil^{\leq} = \{Fam(u)\}$. Hence the redexes in $w/(v/u)$ and s are in different families by [initial], and the lemma follows (since $\simeq_z \subseteq \simeq$).*

The following example shows that a DRS with $\simeq$ and $\hookrightarrow$ relations satisfying all DFS axioms but [initial] need not be stable.

Example 21 Consider the DRS $\mathcal{R}$ given by the figure below, where w and w' are created by u and v, respectively, $u/v = u'$, $v/u = v'$, $w/v' = w'/u' = w^*$. Then the sets $U = \{u, u', v, v'\}$ and $W = \{w, w', w^*\}$ with the contribution relation $U \hookrightarrow W$ do satisfy the DFS axioms except for [initial], but the underlying DRS $\mathcal{R}$ is not stable.



Lemma 22 Let $\mathcal{S}$ be stable, let $t \notin \mathcal{S}$, let $t \xrightarrow{u} t'$, let u be $\mathcal{S}$-unneeded, and let $u' \in t'$ be a redex created by u, in a DFS $\mathcal{F}$. Then u' also is $\mathcal{S}$-unneeded.

Proof By Lemma 9 and Lemma 20.

Now we can generalize the RN theorem, proved in [18,22] for orthogonal ERSs, to all DFSs. In the following we allow for arbitrary stable sets $\mathcal{S}$.

Below $FAM(P)$ denotes the set of families (whose member redexes are) contracted in P.

Theorem 23 (Relative Normalization) Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$, and let $t \notin \mathcal{S}$ be $\mathcal{S}$-normalizable. Then

(1) t contains an $\mathcal{S}$-needed redex.
(2) Any $\mathcal{S}$-needed reduction starting from t eventually terminates at a term in $\mathcal{S}$.

Proof

(1) Let $P : t \twoheadrightarrow s' \to s \xrightarrow{u} e$ be an $\mathcal{S}$-normalizing, and let $s \notin \mathcal{S}$. By the stability of $\mathcal{S}$, u is $\mathcal{S}$-needed. By Lemma 8 and Lemma 22, u is either created by or is a residual of an $\mathcal{S}$-needed redex of s', and (1) follows by repeating the argument.
(2) Let $P : t \twoheadrightarrow s$ be an $\mathcal{S}$-normalizing reduction and $Q : t \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots$ be an $\mathcal{S}$-needed reduction. Further, let $Q_i : t \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots \xrightarrow{u_{i-1}} t_i$ and $P_i = P/Q_i$ ($i \geq 1$) (see the figure below). By $\simeq_z \subseteq \simeq$, $FAM(P_i) \subseteq FAM(P)$ (families are considered with respect to t). Since Q is $\mathcal{S}$-needed and P_i is $\mathcal{S}$-normalizing (by the closure of $\mathcal{S}$ under parallel moves), at least one residual of u_i is contracted in P_i. Therefore, again by $\simeq_z \subseteq \simeq$, $Fam(u_i) \in FAM(P_i)$. Hence $FAM(Q) \subseteq FAM(P)$ and Q is terminating by [FFD].

$$\begin{tikzcd}
t \arrow[r, two heads, "P"] \arrow[d, two heads, "Q_i"'] & s \arrow[d, two heads] \\
t_i \arrow[r, two heads, "P_i"'] \arrow[d, "u_i"'] & \cdot \arrow[d, two heads] \\
t_{i+1} \arrow[r, two heads, "P_{i+1}"'] \arrow[d, two heads] & \cdot \arrow[d, two heads] \\
\cdot \arrow[r, two heads] & \cdot
\end{tikzcd}$$

Note that we have not used the axiom [weak acyclicity] in the proofs. However, it is necessary and sufficient to ensure that the set of normal forms is stable. Note also that only by using Theorem 23 can we prove the analogue of Lemma 13 for all stable $\mathcal{S}$.

Lemma 24 Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$, and let $t \xrightarrow{u} s$. Then any $\mathcal{S}$-needed redex $v \in t$ different from u has an $\mathcal{S}$-needed residual in s.

Proof Same as the proof of Lemma 13, but using Theorem 23 instead of Lemma 12.

We now define *weakly $\mathcal{S}$-needed* redexes, and show that their contraction in an $\mathcal{S}$-normalizable term t leads to an $\mathcal{S}$-normal form of t. We also generalize Lévy's Optimality theorem [51] to all stable sets $\mathcal{S}$ in any DFS.

Definition 25 *We call a family ϕ relative to t $\mathcal{S}$-*needed *if any reduction* *from t to a term in $\mathcal{S}$ contracts at least one member of ϕ. We call redexes in $\mathcal{S}$-needed families* weakly $\mathcal{S}$-needed.

Theorem 26 Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$, and let t be an $\mathcal{S}$-normalizable term in $\mathcal{F}$. Then any weakly $\mathcal{S}$-needed reduction starting from t is terminating.

Proof By [FFD], since there is only a finite number of $\mathcal{S}$-needed families relative to t (that number does not exceed the length of any $\mathcal{S}$-normalizing reduction starting from t).

The above theorem allows one to propagate $\mathcal{S}$-neededness information, obtained from earlier terms, along the reduction, and to contract any residual of an $\mathcal{S}$-needed redex safely (without the danger of missing an $\mathcal{S}$-normal form whenever it exists), even if the residual is no longer $\mathcal{S}$-needed.

Definition 27 A multistep reduction $P : t_0 \longrightarrow\!\!\!\!\rightarrow t_1 \longrightarrow\!\!\!\!\rightarrow \ldots \longrightarrow\!\!\!\!\rightarrow t_n$ is called a *family-reduction* if each $P_i : t_i \longrightarrow\!\!\!\!\rightarrow t_{i+1}$ is a development of a set U_i of redexes belonging to the same family (w.r.t. t_0). $\|P\|$ will denote the number of multisteps in P. The family-reduction P is *complete* if each P_i is the complete development of a maximal set of redexes of t_i belonging to the same family. A family-reduction P is called *$\mathcal{S}$-needed* if each U_i contains at least one $\mathcal{S}$-needed redex.

Lemma 28 *Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$. Then any sequentialization of an $\mathcal{S}$-needed family-reduction is weakly $\mathcal{S}$-needed.*

Proof *Let $P : t_0 \xrightarrow{U_0}\!\!\!\!\rightarrow t_1 \xrightarrow{U_1}\!\!\!\!\rightarrow \ldots \xrightarrow{U_{k-1}}\!\!\!\!\rightarrow t_k$ be an $\mathcal{S}$-needed family-reduction, and let $u_i \in U_i$ be $\mathcal{S}$-needed ($i = 0, \ldots, k-1$). We need to show that U_i (with history $P_i : t_0 \xrightarrow{U_0}\!\!\!\!\rightarrow \ldots \xrightarrow{U_{i-1}}\!\!\!\!\rightarrow t_i$) is an $\mathcal{S}$-needed family, i.e., any $\mathcal{S}$-normalizing reduction Q starting from t_0 contracts at least one redex in the family of U_i. By closure of $\mathcal{S}$ under reduction, Q/P_i is $\mathcal{S}$ normalizing (for every i), and by $\mathcal{S}$-neededness of u_i, Q/P_i contracts at least one residual of u_i, thus Q indeed contracts a redex in the family of U_i.*

Corollary 29 *Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$. Then any $\mathcal{S}$-needed family-reduction starting from an $\mathcal{S}$-normalizable term is eventually $\mathcal{S}$-normalizing.*

Lemma 30 (Unique Families) Every family is contracted at most once in a complete family-reduction.

Proof Let $P_n : t_0 \xrightarrow{U_0}\!\!\!\!\rightarrow t_1 \xrightarrow{U_1}\!\!\!\!\rightarrow \ldots \xrightarrow{U_{n-1}}\!\!\!\!\rightarrow t_n$ be a complete family-reduction. We show by induction on $n = \|P\|$ that $(a)_n$: all families contracted in P_n are different; and $(b)_n$: there is no redex in t_n whose family has been contracted in

P_n. The case $n = 0$ is clear. Further, $(a)_n$ follows immediately from $(a)_{n-1}$ and $(b)_{n-1}$. Again by $(a)_{n-1}$ and $(b)_{n-1}$, and by the completeness of P_n, all redexes in t_n that are residuals of redexes of t_{n-1} are in families that have not been contracted before. By [creation], for the family ϕ of a created redex in t_n, we have $Fam(U_{n-1}) \hookrightarrow \phi$; by $(a)_{n-1}$, $(b)_{n-1}$, and [contribution], $Fam(U_{n-1}) \not\hookrightarrow Fam(U_i)$, for any $i \leq n-1$. Hence $\lceil Fam(U_i) \rceil^{\leq} \neq \lceil \phi \rceil^{\leq}$, and $(b)_n$ follows.

Theorem 31 (Relative Optimality) Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$, and let t be an $\mathcal{S}$-normalizable term in $\mathcal{F}$. Then any $\mathcal{S}$-needed $\mathcal{S}$-normalizing complete family-reduction $Q : t \twoheadrightarrow e \in \mathcal{S}$ is $\mathcal{S}$-optimal in the sense that it has the minimal number of family-reduction steps.

Proof As in the λ-calculus [51]. Let $P : t \twoheadrightarrow s$ be an $\mathcal{S}$-normalizing family-reduction. It follows from the proof of Theorem 23 that $FAM(Q) \subseteq FAM(P)$ (see the figure for Theorem 23). Hence, by Lemma 30, $\|Q\| = n(FAM(Q)) \leq n(FAM(P)) \leq \|P\|$, where $n(FAM(Q))$ denotes the number of families in $FAM(Q)$.

5 Equivalence of Zig-zag and Extraction in Affine SDRSs

We will use the term *Affine* Stable Deterministic Residual Structure (ASDRS) for an SDRS with a non-duplicating residual relation. In this section, we introduce an abstract extraction algorithm for ASDRSs and show that zig-zag related redexes (with histories) have the same canonical representatives w.r.t. extraction, up to an equivalence on histories. These canonical representatives are obtained as normal forms of redexes with histories Pv w.r.t. the extraction procedure, which eliminates all steps of histories P that do not 'contribute' to the family of v. [3]

Lévy introduced an extraction procedure for the λ-calculus in [50,51] in order to prove decidability of the family relation. His extraction procedure is effective, and gives canonical representatives of families, which are unique, thus implying the decidability of the family relation. For higher order rewrite systems, whose reduction steps are more complicated, there are two conceptually different extensions of Lévy's extraction algorithm. The first is due to Asperti and Laneve [4], and the second to van Oostrom [62].

A 'problem' arises because a redex can create a number of redexes 'intuitively' in the same family without the help of previous steps, something which cannot

[3] The extraction process was thought to require the syntactic structure of terms. To quote Lévy [51]: 'We turn now to the hard part of this paper, which is to show that the family relation is decidable. The trouble comes from the *necessity* of looking now inside λ-expressions and from not being able to go on with algebraic manipulations'.

happen in the λ-calculus or term rewriting. That these redexes are intuitively in the same family, can be seen after decomposing the rewrite step into two parts – the *TRS part* that only creates new symbols, and the *substitution part*, that performs all (often nested) substitutions. The substitution part can duplicate or erase the redexes created during the TRS part, and all substitution copies of a redex created by the TRS-part are viewed as belonging to the same family, as the labels of such redexes are the same. Such redexes cannot be related by the zig-zag if one works with the original system [4], but can be related if one works in the refinement of the original system [62]. Now, the difference between the two approaches is that Asperti and Laneve decided to accept the inadequacy of the zig-zag, but extended Lévy's extraction algorithm by the *shift* operation which relates all copies of the simultaneously created redexes of the same family to a canonical representative, thus making extraction match the labelling; the resulting family-relation is *non-separable* – a redex can create multiple members of the same family simultaneously. On the contrary, van Oostrom works with the refined rewrite systems and no operation like shift is necessary to ensure coincidence of labelling, extraction and zig-zag families. Since we want to define an extraction procedure adequate for zig-zag, we do not need an operation modelling *shift*. Our results in Section 7 will shed further light on the separability problem.

Another relevant extraction procedure is studied by Kennaway et al. [33] for orthogonal Term Graph Rewriting Systems (TGRSs). That paper is concerned with Event Structure [70,58,71] semantics for orthogonal TGRSs and does not mention families explicitly : pre-events there correspond to redexes with histories. The extraction algorithm in [33], called there the *minimization* algorithm, is based on a syntactic concept of the contribution relation on redexes and is technically different from our extraction algorithm. Note that TGRSs are non-duplicating rewrite systems, hence our extraction algorithm, as well as other results in this paper concerning ASDRSs, apply to orthogonal TGRSs.

We start by studying standard reductions in ASDRSs. Our standardization concept differs from the usual 'left-to-right and outside-in' concept of standardization in the λ-calculus [16]. Our standard reductions are simply 'self-needed' reductions – reductions in which every step essentially contributes to the whole computation – since we do not have any nesting relation imposed on redexes, unlike in the ARSs of [24], and there is no concept of 'left' or 'right' occurrences in DRSs. However, our concept of standardization captures the essence of the usual notion in many respects. In particular, in the extraction process which we study below, self-needed reductions play the same role as left-to-right outside-in standard reductions in the extraction processes of [51,4,62]. For a theory of standardization for stable DRSs in general, see [39,40]. And for an 'outside-in theory' of standardization in ARSs with an axiomatized nesting relation, see [24,54].

Definition 32 • *Let $P : t \twoheadrightarrow o$ and $u \in t$, in a DRS. We call u* erased *in P or P-*erased *if $u/P = \emptyset$. We say that P* discards *u if P is external to u and erases it.*

- *We call u P-*needed *if there is no $Q \approx_L P$ that is external to u, and call it P-*unneeded *otherwise.*

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

- *Let $Q : t \twoheadrightarrow o$, $P : t \overset{P'}{\twoheadrightarrow} s \twoheadrightarrow e$, and $u \in s$. We say that u is Q-needed, or more precisely, $P'u$ is Q-needed, if u is Q/P'-needed. We call P Q-*needed *if every redex contracted in P is. We call P* self-needed *or* standard *if it is P-needed. The other concepts above are extended in the same way.*

Note that P-neededness and P-erasure do not depend on the choice of a reduction in the class $\langle P \rangle_L$ of reductions Lévy-equivalent to P, since $u/P = u/Q$ when $P \approx_L Q$. The *external* and *discards* concepts however do depend on the particular reduction in the Lévy-equivalence class.

Lemma 33 *Let $P : s \twoheadrightarrow t \xrightarrow{u} e \overset{P'}{\twoheadrightarrow} o$ in an ASDRS.*

(1) *If $P : s \twoheadrightarrow s' \xrightarrow{w} o$, then $w \in s'$ is P-needed.*
(2) *If u creates $v \in e$ and is P-unneeded, then so is v.*
(3) *If $u \neq v \in t$, then v is P-needed iff it has a P-needed residual in e.*

Proof

(1) *By Lemma 5 (Weak Acyclicity Lemma), there is no $Q : s' \twoheadrightarrow o$ such that Q is external to w and $Q \approx_L w$. Hence w is w-needed, i.e., is P-needed.*
(2) *Since u is P-unneeded and P-erased, there is $Q_t \approx_L u+P'$ that discards u. By Lemma 6 (Stability Lemma), $Q_e = Q_t/u$ is external to v, and $Q_e \approx_L P'$. Hence v is P-unneeded.*
(3) ***(⇒)*** *If on the contrary $v/u = \emptyset$, then $u + P'$ would discard v, contradicting P-neededness of v. So let v' be the only u-residual of v. 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 – a contradiction.*
(⇐) *Let v have a P-needed residual $v'' \in e$. If on the contrary v is P-unneeded, there is $Q_t \approx_L u + P'$ that is external to v. Hence $u/Q_t = \emptyset$ (since $u/(u+P') = \emptyset$) and $Q_e = Q_t/u \approx_L P'$. By Lemma 4, Q_e is external to v'', contradicting P-neededness of v''.*

The following is a standardization procedure for reductions in ASDRSs.

Definition 34 *Let $P : t \twoheadrightarrow s$. The* canonical standard variant of *P, $ST(P)$, is defined as follows: If $P = \emptyset$, then so is $ST(P)$. Otherwise, let $v \in t$ be such that it is P-needed and its residual is contracted in P first among P-needed residuals of P-needed redexes in t (existence of such v follows from Lemma 33).*

Then $ST(P) = v + ST(P/v)$.

Theorem 35 (Standardization) *For any finite reduction P in an ASDRS, $ST(P)$ is a finite standard reduction Lévy-equivalent to P.*

Proof *Termination of the standardization follows immediately from the fact that, in the above definition, $|P/v| \leq |P| - 1$, and we have immediately from the construction that $P \approx_L v + P/v \approx_L \ldots \approx_L ST(P)$.*

We write $P \approx_{STA} Q$ if $P \approx_L Q$ and both P and Q are standard. For any standard P, we define $\langle P \rangle_S = \{Q \,|\, Q \approx_{STA} P\}$. Further, we write $Q \in STV(P)$ if $Q \in STA$ and $Q \approx_L P$, where STA denotes the set of all standard reductions, and call Q a *standard variant* of P. Note that Lemma 33 gives an algorithm of construction of a standard variant of any finite reduction $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots \xrightarrow{u_{n-1}} t_n$ in an ASDRS. Indeed, the last step u_{n-1} of P is P-needed by Lemma 33.(1). If it is created by u_{n-2}, then the latter is P-needed too, by Lemma 33.(2). Otherwise, the ancestor redex of u_{n-1} in t_{n-2} is P-needed by Lemma 33.(3). Similarly, we can trace the 'responsible' redex of u_{n-1} in t_0, which is P-needed. Repeated contraction of P-needed redexes terminates, and yields a standard variant of P; the proof is similar to the proof of Theorem 35. The next proposition shows that, moreover, all standard variants of a finite reduction P in an ASDRS can be found effectively. (Since the length of any standard variant of P coincides with the number of P-needed steps in P, P has only a finite number of standard variants.)

Proposition 36 *For any finite reduction P in an ASDRS, P-neededness of redexes in all terms of P is decidable. Consequently, any standard variant of P can be constructed efficiently (in particular, the standardization procedure of P is computable).*

Proof *Let $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots \rightarrow t_n$. P-(un)neededness of any redex in a term t_i can be established by induction on $n = |P|$, as follows. If $n = 1$, then only the contracted redex u_0 is P-needed in t_0 by Definition 32 and Lemma 33.(1). Let $n > 1$, and let $P_1 : t_1 \xrightarrow{u_1} t_2 \xrightarrow{u_2} \ldots \rightarrow t_n$. We can assume to have found all the P-needed redexes in all t_i with $i \geq 1$, since P-neededness in these terms coincides with P_1-neededness by Definition 32 (and $|P_1| = n - 1$). Then a redex in t_0 different from u_0 is P-needed iff it has such a residual in t_1, by Lemma 33.(3). If u_0 does not create u_1, then they can be permuted. If $u_1 = u_1'/u_0$ and $u_0' = u_0/u_1' = \emptyset$, then u_0 is P-unneeded by Definition 32; otherwise, if $u_0' \neq \emptyset$, by the induction assumption, we can assume to know whether or not u_0' is needed w.r.t. $P' = u_0' + u_2 + \ldots + u_{n-1}$, and u_0 is P-needed iff u_0' is P'-needed. Finally, if u_0 creates u_1, we can standardize P_1, or construct a standard variant P_1' of P_1; then if u_0 still creates the first step of P_1' (which is P-needed by Definition 32), then u_0 is P-needed by Lemma 33.(2); if not, then we arrive to a previously considered case, and the decidability of*

P-neededness follows. The rest is immediate from the discussion above.

Definition 37 *Let $P : t \twoheadrightarrow s$ in an ASDRS, and let $v \in s$. We call Pv* standard *if so is P. We call Pv* canonical *if it is standard and, for any $Q \approx_L P$, the last step in Q creates v.*

Note that if $P \approx_{STA} P'$, then Pv is canonical iff so is $P'v$. So the canonical forms we speak of are actually objects $\langle P \rangle_S v$, for standard finite reductions P. Our extraction algorithm, defined in Definition 39 below, transforms any redex with history into a canonical one, and the main result of this section is that redexes are zig-zag related iff they have the same canonical form.

The 'syntactic' extraction procedures in [51,4,62] (without the *shift* operation) can quickly be summarized as follows: First, given a Pu, one must standardize $P + u$ according to a left-to-right outside-in standardization procedure, and if $P' + u' + P''$ is the standard variant of $P + u$, where u is a P''-residual of u', one starts the extraction procedure from $P'u'$. Now, if the last step, say w, of P' does not create u', then the contraction of w can be 'omitted' in P'. Otherwise there is a non-empty tail P'_2 of P', say $P' = P'_1 + v + P'_2$, that takes place either in a part disjoint from v or inside a descendant of an argument of v, and in both cases v does not contribute to the creation of u', therefore is 'omitted' by the extraction step. Since redexes in DRSs do not have any structure (in particular, they are not partitioned into patterns and arguments), these 'syntactic' extraction algorithms (and corresponding correctness proofs) cannot be directly generalized to ASDRSs. The same is true for the extraction algorithm in [33]. Instead, our extraction algorithm (when applied to Pu) checks every standard variant of P in order to find one whose last step does not create u and therefore can be omitted. The following lemma says that we can indeed restrict our search to standard reductions only:

Lemma 38 *Let $Q : t \overset{P}{\twoheadrightarrow} s \overset{u}{\to} e$, where u does not create $v \in e$. Then there is a standard $Q' \approx_L Q$ such that $Q' : t \overset{P'}{\twoheadrightarrow} s' \overset{u'}{\to} e$, where $P'u' \trianglelefteq_z Pu$ and u' does not create v.*

Proof *We show that $ST(Q)$ can be taken for Q'. By Definition 34, $ST(Q)$ is obtained from Q by a sequence of transformations $Q = Q_1, Q_2, \ldots, Q_n = ST(Q)$ such that Q_{i+1} is obtained from Q_i by permuting the first Q-needed step that has preceding Q-unneeded steps before those Q-unneeded steps (all Q_i are Lévy-equivalent). Since u is the last Q-needed step in Q by Lemma 33.(1), any Q_i with $i < n$ has the form $P_i + u$ such that $P_i \approx_L P$, and P_{n-1} has the form $P_{n-1} : t \overset{P'}{\twoheadrightarrow} o \overset{P''}{\twoheadrightarrow} s$ where P' is Q-needed and P'' is Q-unneeded. By Lemma 33.(2), P'' cannot create u, i.e., there is $u' \in o$ such that $u'/P'' = u$, and u' is Q-needed by Lemma 33.(3) (see the diagram). Since P''/u' is Q-unneeded by Lemma 33.(3), and since the last step of $P' + u' + P''/u'$ is Q-needed by Lemma 33.(1), $P''/u' = \emptyset$. Since u' is Q-needed and P'' is*

Q-unneeded, P'' is external to u' by Lemma 33.(3). Hence, by the Stability Lemma, u' does not create v, and the lemma follows since $ST(Q) = P' + u'$ is standard by Theorem 35, and $P'u' \trianglelefteq_z Pu$ since $u = u'/P''$.

$$
\begin{tikzcd}
t \arrow[r, two heads, "P'"] & o \arrow[r, two heads, "P''"] \arrow[d, "u'"'] & s \arrow[r, "u"] \arrow[d, "u"] & e \ni v \arrow[d, two heads, "\emptyset"] \\
 & v \in e \arrow[r, two heads, "\emptyset"'] & e \ni v \arrow[r, two heads, "\emptyset"'] & e \ni v
\end{tikzcd}
$$

It follows from Definition 37, Proposition 36 and Lemma 38 that it is decidable whether a redex with history Pv is canonical, and if Pv is not canonical, one can effectively find a reduction $Q : t \overset{P'}{\twoheadrightarrow} e \overset{u}{\rightarrow} s$ such that $Q \in STV(P)$ and u does not create v, say $v = v'/u$. In the search for a shortest reduction that creates a redex in the zig-zag class of Pv, contraction of u in Q can be omitted – $P'v' \simeq_z Pv$ and $|P'| < |P|$, since all standard Lévy-equivalent reductions have the same (minimal) length. Obviously, reductions creating a redex in some family in the quickest way must be standard, since they are the shortest in their Lévy-equivalence classes. The transformation of Pv into $P'v'$ is denoted by $Pv \overset{u}{\rightharpoonup} P'v'$, or just $Pv \rightharpoonup P'v'$. For example, consider the ASDRS corresponding to the TRS $R = \{f(x) \to g(x),\, a \to b,\, b \to c\}$, let $P : f(a) \to g(a) \to g(b)$, and let $v = b$ in $g(b)$. Then Pv is not canonical, as $P \approx_{STA} Q : f(a) \to f(b) \to g(b)$ and the last step of Q does not create v – the latter is the residual of $v' = b$ in the final term of $P' : f(a) \to f(b)$. Hence we can perform an extraction step $Pv \rightharpoonup P'v'$. The latter redex is in extraction normal form.

The formal definition of extraction is as follows:

Definition 39 (Extraction) *Let $Q : t \overset{P'}{\twoheadrightarrow} e \overset{u}{\rightarrow} s$ be a standard variant of P, in an ASDRS, and let $v \in s$ be a u-residual of $v' \in e$. Then we write $Pv \overset{u}{\rightharpoonup} P'v'$, and call the transformation an* extraction *step. (Note that, since Q is standard, so is P' by Definition 32.) $\rightharpoonup\!\!\!\!\rightharpoonup$ is the transitive and reflexive closure of $\rightharpoonup$.*

Since in the above definition $|P'| < |Q| \leq |P|$, the relation $\rightharpoonup$ is trivially strongly normalizing, and in order to prove that it is confluent (modulo $\approx_{STA}$ on histories), it is enough to prove that it is weakly (or locally) confluent, that is, $Qw'' \overset{v}{\leftharpoondown} Nw \overset{u}{\rightharpoonup} Pw'$ implies $Qw'' \overset{u'}{\rightharpoonup} N^*w^* \overset{v'}{\leftharpoondown} Pw'$. This will be proved in Theorem 42. We need two lemmas first, which state particular cases of a general fact that causally unrelated steps (or rather families) can be contracted in any order.

Lemma 40 *Let $u + P \approx_{STA} Q + u'$, where $u' = u/Q$ and $P = v + P'$. Then u does not create v, and u can be contracted after v, i.e., $u + v \approx_L v^* + u^*$,* *where $v = v^*/u$ and $u^* = u/v^*$. Further, $v^*/Q = \emptyset$ and $u' = u^*/(Q/v^*)$.*

Proof *Let $Q' = Q/u$. Since $u + P$ is standard, so is P by Definition 32, so v is P-needed, and since $P \approx_L Q'$, v is Q-needed too, i.e, Q' contracts a residual v' of v. Since Q' contracts residuals of redexes contracted in Q, Q contracts a redex v'' whose residual is v'. So we have the following picture:*

$$\begin{tikzcd}
\cdot \arrow[r, "u"] \arrow[d] & \cdot \arrow[r, "v"] \arrow[d] & \cdot \arrow[r, two heads, "P'"] \arrow[d] & \cdot \arrow[d] \\
\cdot \arrow[r] \arrow[d, "v''"'] & \cdot \arrow[r, "v'"] \arrow[d, "v'"'] & \cdot \arrow[r, two heads] \arrow[d] & \cdot \arrow[d] \\
\cdot \arrow[r, two heads] \arrow[d] & \cdot \arrow[r, two heads] \arrow[d] & \cdot \arrow[r, two heads] \arrow[d] & \cdot \arrow[d] \\
Q \arrow[r, "u'"'] & Q' \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads] & \cdot
\end{tikzcd}$$

Now, since Q is external to u (since u has a Q-residual u' and the DRS is non-duplicating), we have immediately by the Stability Lemma that both v and v'' are residuals of some redex v^* in the initial term. Hence $u+v \approx_L v^*+u^*$, where $v = v^*/u$ and $u^* = u/v^*$. Since Q contracts a residual v'' of v^*, $v^*/Q = \emptyset$. Since Q is external to u, we have by Lemma 4 that $Q^* = Q/v^*$ is external to u^*. Since u is $u + P$-needed, so is u^* by Lemma 33.(3). Hence u^* is $Q^* + u'$-needed. Since Q^* is external to u^* and $Q^* + u'$ contracts a residual of u^*, we have $u' = u^*/Q^*$.

$$\begin{tikzcd}
\cdot \arrow[r, "v^*"] \arrow[d, "Q"'] & \cdot \arrow[r, "u^*"] \arrow[d, "Q^*"'] & \cdot \arrow[r, two heads, "P'"] \arrow[d] & \cdot \arrow[d] \\
\cdot \arrow[r, two heads, "\emptyset"] \arrow[d, "u'"'] & \cdot \arrow[r, "u'"] \arrow[d, "u'"'] & \cdot \arrow[r, two heads] \arrow[d, "\emptyset"] & \cdot \arrow[d] \\
\cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads] & \cdot
\end{tikzcd}$$

Lemma 41 *Let $P + u \approx_{STA} Q + v$ and let $u \neq v$. Then there are $P'v'$ and $Q'u'$ such that $P' + v' \approx_{STA} P$, $Q' + u' \approx_{STA} Q$, $P' \approx_{STA} Q'$, $P'v' \simeq_z Qv$ and*

$Q'u' \simeq_z Pu$, $u = u'/v'$ and $v = v'/u'$.

$$\begin{tikzcd}
\cdot \arrow[r, two heads, "P'"] \arrow[d, "Q'"'] & \cdot \arrow[r, "v'"] \arrow[d, "\emptyset"] & \cdot \arrow[r, "u"] \arrow[d, "\emptyset"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, two heads, "\emptyset"] \arrow[d, "u'"'] & \cdot \arrow[r, "v'"] \arrow[d, "u'"] & \cdot \arrow[r, "u"] \arrow[d, "u"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, two heads, "\emptyset"'] \arrow[d, "v"'] & \cdot \arrow[r, "v"] \arrow[d, "v"'] & \cdot \arrow[r, two heads, "\emptyset"'] \arrow[d, "\emptyset"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot
\end{tikzcd}$$

Proof *Since $P + u \approx_{STA} Q + v$, we have $v \approx_L (P + u)/Q$. By Lemma 5, $(P + u)/Q$ contracts a residual of v. Note that $P \approx_L Q$ would imply $u \approx_L v$, implying by [weak acyclicity] that $u = v$. Thus $P \not\approx_L Q$. We show that $P/Q \neq \emptyset$.*

$$\begin{tikzcd}
\cdot \arrow[r, two heads, "P"] \arrow[d, "Q"'] & \cdot \arrow[r, "u"] \arrow[d, "Q/P"', "u"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, two heads, "P/Q", "v"'] \arrow[d, "v"'] & \cdot \arrow[r, two heads, "\emptyset"'] \arrow[d] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot
\end{tikzcd}$$

Suppose on the contrary that $P/Q = \emptyset$. Then, $Q/P \neq \emptyset$ and by [weak acyclicity], $v = u/(Q/P)$. But $P + u \approx_L Q + v$ implies $(Q/P)/u = \emptyset$. Since $Q + v$ is standard, the first (and any other) step of Q whose residual, say w, is contracted in Q/P is u-needed by Lemma 33.(3). Hence $w/u = \emptyset$ implies $u = w$, and therefore $Q/P = u$ and $u/(Q/P) = \emptyset$, contrary to $v = u/(Q/P)$. So $P/Q \neq \emptyset$. Since P is $P+u$-needed (recall that $P+u$ is standard), so is P/Q, i.e., P/Q is v-needed. Hence, by [weak acyclicity], the first (and the only) step of P/Q coincides with v, i.e., $P/Q = v$. Thus P contracts a redex v'' whose residual is v. So if $P = P_1 + v'' + P_2$, then $(Q + v)/P_1 = Q/P_1 + v$, and we have $v'' + P_2 + u \approx_{STA} Q/P_1 + v$ and $v'' + P_2 \not\approx_L Q/P_1$ (see the figure).

$$\begin{tikzcd}
\cdot \arrow[r, two heads, "P_1"] \arrow[d, "Q"'] & \cdot \arrow[r, "v''"] \arrow[d] & \cdot \arrow[r, two heads, "P_2"] \arrow[d] & \cdot \arrow[r, "u"] \arrow[d] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, two heads, "\emptyset"] \arrow[d, "v"'] & \cdot \arrow[r, "v"] \arrow[d, "v"'] & \cdot \arrow[r, two heads, "\emptyset"] \arrow[d, "\emptyset"] & \cdot \arrow[r, two heads, "\emptyset"] \arrow[d, "\emptyset"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot
\end{tikzcd}$$

Now, by repeated application of Lemma 40, $v'' + P_2 + u$ can be transformed into a reduction $P_2' + v' + u$ such that $v'' + P_2 \approx_{STA} P_2' + v'$, $v' = v''/P_2'$, and $v = v'/(Q/(P_1 + P_2'))$.

$$\begin{tikzcd}
\cdot \arrow[r, "P_1"] \arrow[d, "Q"'] & \cdot \arrow[r, twoheadrightarrow, "P_2'"] \arrow[d] & \cdot \arrow[r, "v'"] \arrow[d] & \cdot \arrow[r, "u"] \arrow[d] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, twoheadrightarrow, "\emptyset"] \arrow[d, "v"'] & \cdot \arrow[r, twoheadrightarrow, "\emptyset"] \arrow[d, "v"'] & \cdot \arrow[r, "v"] \arrow[d, "v"'] & \cdot \arrow[r, twoheadrightarrow, "\emptyset"] \arrow[d] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, twoheadrightarrow, "\emptyset"'] & \cdot \arrow[r, twoheadrightarrow, "\emptyset"'] & \cdot \arrow[r, twoheadrightarrow, "\emptyset"'] & \cdot \arrow[r, twoheadrightarrow, "\emptyset"'] & \cdot
\end{tikzcd}$$

Hence, if we take $P' = P_1 + P_2'$, we have that $P' + v' \approx_{STA} P$ and $P'v' \trianglelefteq_z Qv$. Existence of $Q'u'$ such that $Q' + u' \approx_{STA} Q$ and $Q'u' \trianglelefteq_z Pu$ can be shown similarly. Since $P_2/(Q/(P_1 + v'')) = \emptyset$, we have again by repeated application of Lemma 40 that $P'/Q \approx_L P'/(Q' + u') \approx_L (P'/Q')/u' = \emptyset$. But P'/Q' is external to u' since u' has a $P/Q' \approx_L P'/Q' + v'/(Q'/P')$-residual (by $Q'u' \trianglelefteq_z Pu$).

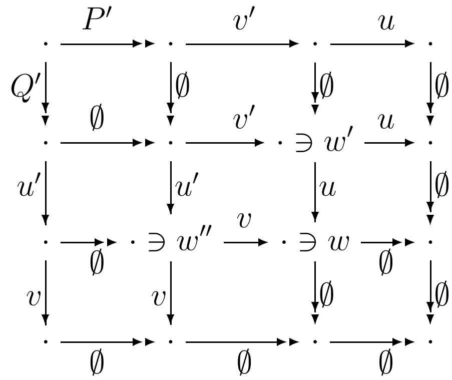
$$\begin{tikzcd}
\cdot \arrow[r, twoheadrightarrow, "P'"] \arrow[d, "Q'"'] & \cdot \arrow[r, "v'"] \arrow[d, "\emptyset"] & \cdot \arrow[r, "u"] \arrow[d, "\emptyset"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, twoheadrightarrow, "\emptyset"] \arrow[d, "u'"'] & \cdot \arrow[r, "v'"] \arrow[d, "u'"] & \cdot \arrow[r, "u"] \arrow[d, "u"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, twoheadrightarrow, "\emptyset"'] \arrow[d, "v"'] & \cdot \arrow[r, "v"] \arrow[d, "v"'] & \cdot \arrow[r, twoheadrightarrow, "\emptyset"'] \arrow[d, "\emptyset"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, twoheadrightarrow, "\emptyset"'] & \cdot \arrow[r, twoheadrightarrow, "\emptyset"'] & \cdot \arrow[r, twoheadrightarrow, "\emptyset"'] & \cdot
\end{tikzcd}$$

Hence we have by Lemma 5 and Lemma 33.(3) that $P'/Q' = \emptyset$ (as otherwise P'/Q' should have a non-empty u'-residual by Lemma 33.(3)). The converse is proved similarly, so $P' \approx_L Q'$. It follows that $u = u'/v'$ and $v = v'/u'$.

Theorem 42 (Extraction) *Every redex Pv in an ASDRS has exactly one extraction normal form $\langle P^* \rangle_S v^*$ which is canonical, and $P^*v^* \trianglelefteq_z Pv$.*

Proof *It is enough to show that the extraction relation $\rightharpoondown$ is weakly confluent (as it is strongly normalizing). So let $Qw'' \overset{v}{\leftharpoondown} Nw \overset{u}{\rightharpoondown} Pw'$ with $u \neq v$ (since if $u = v$ then there is nothing to prove). We will show that $Qw'' \overset{u'}{\rightharpoondown} N^*w^* \overset{v'}{\leftharpoondown} Pw'$ for some N^*w^*, u', and v' such that $u = u'/v'$ and $v = v'/u'$. By Definition 39, we have from $Qw'' \overset{v}{\leftharpoondown} Nw \overset{u}{\rightharpoondown} Pw'$ that $Q + v \approx_{STA} N' \approx_{STA} P + u$, where N' is a standard variant of N, and $w''/v = w'/u = w$. By Lemma 41, we have the following situation, where $P' + v' \approx_{STA} P$, $Q' + u' \approx_{STA} Q$, $P' \approx_{STA} Q'$,*

$u = u'/v'$, and $v = v'/u'$ (hence $P'v' \simeq_z Qv$, $Q'u' \simeq_z Pu$).



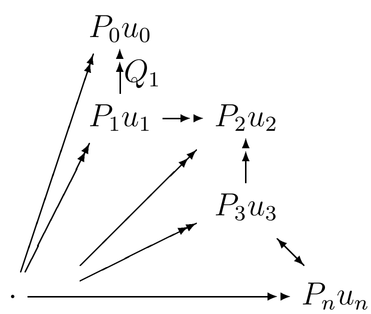
Now, by [stability], there is a redex w^* in the final term of P' (and Q') such that $w^*/v' = w'$ and $w^*/u' = w''$. Thus, for $N^* = P'$, we have $Qw'' \overset{u'}{\rightharpoondown} N^*w^* \overset{v'}{\leftharpoondown} Pw'$ by Definition 39.

The extraction normal form $\langle P^* \rangle_S v^*$ of Pv is called a *canonical form of* Pv, and so are all $P'v' \in \langle P^* \rangle_S v^*$. Now we can prove the adequacy of our extraction procedure for the zig-zag.

Theorem 43 *In an ASDRS, $Pu \simeq_z Qv$ iff they have the same unique canonical form $\langle N \rangle_S w$.*

Proof

($\Rightarrow$) By definition of $\simeq_z$, there are $P_0u_0 = Pu, P_1u_1, \ldots, P_nu_n = Qv$ such that $P_0u_0 \trianglerighteq_z P_1u_1 \trianglelefteq_z P_2u_2 \trianglerighteq_z \ldots P_nu_n$. By the Standardization Theorem, we can take P_i to be standard.



Since $P_0u_0 \trianglerighteq_z P_1u_1$, there is Q_1 such that $P_0 \approx_L P_1 + Q_1$ and $u_0 = u_1/Q_1$. Let $P_1'u_1'$ be a canonical form of P_1u_1: $P_1u_1 \rightharpoondown P_1'u_1'$. Then there is P_1^* such that $P_1 \approx_{STA} P_1' + P_1^*$. We show that P_1' is $P_1' + P_1^* + Q_1$-needed, i.e., P_0-needed (since $P_1' + P_1^* + Q_1 \approx_L P_0$). Suppose on the contrary that P_1' contracts a P_0-unneeded redex. Let w be the latest P_0-unneeded step in P_1'. By Lemma 33.(2), w does not create the next step in P_1' (if w is not the last step in P_1'), therefore it can be permuted with its next step. That w-step is again P_1'-unneeded by Lemma 33.(3), and can be contracted after its next *step, and so on. So we can assume that w is the last step in P_1' (P_1' is chosen up to $\approx_{STA}$). Thus there is $N \approx_L w + P_1^* + Q_1$ that is external to w, and $w/N = \emptyset$. Since $P_1'u_1'$ is canonical, w creates u_1'. Hence, by the Stability Lemma, $\emptyset = u_1'/(N/w) = u_1'/(P_1^* + Q_1) = u_1$ – a contradiction. Thus P_1' is P_0-needed. This implies that the standardization procedure of Definition 34 does not effect P_1' when applied to $P_1' + P_1^* + Q_1$, i.e., we can assume a standard $P_0' \approx_{STA} P_0$ such that $P_0' = P_1' + P_0''$ for some P_0'', and $u_0 = u_1'/P_0''$. Hence $P_0u_0 \rightharpoonup\!\!\!\rightharpoonup P_1'u_1'$ by the definition of $\rightharpoonup$, and $P_1'u_1'$ is a canonical form of both P_0u_0 and P_1u_1. Similarly, since $P_1u_1 \trianglelefteq_z P_2u_2$, we have that $P_1'u_1'$ is a canonical form of P_2u_2, and so on. Now it is enough to apply Theorem 42.*

($\Leftarrow$) Immediate from Theorem 42 and the definition of $\simeq_z$.

This theorem, together with computability of extraction normal forms, implies decidability of the zig-zag relation in ASDRSs. Note that, at this stage, we do not yet know whether or not zig-zag is a family relation. This is the subject of the next section.

6 Affine Zig-Zag and Separable Families

In this section we show that, in ASDRSs, the zig-zag relation forms a family relation, that is, it satisfies the family axioms of DFSs. We will also define *separability* of a family relation, and show that zig-zag is the only separable family relation in ASDRSs.

Definition 44 *(1) We call a DFS $\mathcal{F}$ a* zig-zag *DFS, ZDFS, if its family relation is the zig-zag $\simeq_z$.*

(2) We call an affine DFS separable*, SDFS, if, for any redex Pv, v cannot create two different redexes in the same family, that is, if v creates w', w'' and $w' \neq w''$, then $Fam((P+v)w') \neq Fam((P+v)w'')$. In a separable DFS, the family and contribution relations will be denoted by $\simeq_s$ and $\hookrightarrow_s$. The DFS is* non-separable *otherwise.*

It is easy to see that not all (even linear, i.e., without duplication and erasure) DFSs are zig-zag DFSs or separable DFSs. For example, in the linear SDRS given by the reduction graph

$$\begin{tikzcd}
 & & \cdot \arrow[dr, "v_2'"] & \\
\cdot \arrow[r, "u"] & \cdot \arrow[ur, "v_1"] \arrow[dr, "v_2"'] & & \cdot \\
 & & \cdot \arrow[ur, "v_1'"'] &
\end{tikzcd}$$

where u creates v_1 and v_2, $v_1' = v_1/v_2$ and $v_2' = v_2/v_1$, we can set all vs to belong to the same family ϕ, and u to form its own family ψ, and define $\psi \hookrightarrow \phi$. Then we get a DFS which is not a ZDFS as for example $uv_1 \not\simeq_z uv_2$. That DFS is not separable either, as u creates two different members of the family ϕ. We will show below that this is not a coincidence. We could also define $\{v_1, v_1'\}$ and $\{v_2, v_2'\}$ to form separate families ϕ_1 and ϕ_2, and define $\psi \hookrightarrow \phi_1, \phi_2$, and this would yield a ZDFS.

As already mentioned in the introduction, non-separable families (not using that name) for Interaction Systems are studied in [4], where it is demonstrated that such a family relation is in general strictly larger than the zig-zag. Indeed, consider the following example from [4]. Let an Interaction System be given by the μ-rule: μ-rule $\mu(\lambda x.X) \to [\mu(\lambda x.X)/x]X$. Then the μ-reduction step $t = \mu(\lambda x.(xx)) \to (\mu(\lambda x.(xx))\,\mu(\lambda x.(xx))) = s$ simultaneously creates the two μ-redexes in s; these redexes are intuitively in the same family (and this intuition is supported by an appropriate labelling system for Interaction Systems), but cannot be related by zig-zag (without taking into account the structure of rewrite steps, which is impossible to do for DRS steps as they do not carry any internal structure).

Another important example where a non-separable sharing is reasonable is the *lazy call-by-value* λ-calculus, λ_{LV} [64], which is obtained from the λ-calculus by allowing only β-redexes whose arguments are *values* (i.e, variables or abstractions $\lambda x.t$), and that are not in the scope of λ-occurrences (we assume that there are no δ-rules in the calculus). it is easy to see that λ_{LV} is linear: 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} (i.e., that the residuals of redexes remain admissible) follows from a similar argument. Now consider the reduction

$$t = w = (\lambda x.(xy)(xy))\lambda x.u \xrightarrow{w} o = \overbrace{((\lambda x.u)y)}^{v_1}\overbrace{((\lambda x.u)y)}^{v_2} \twoheadrightarrow uu = s,$$

where u is a λ_{LV}-redex such that x does not occur free in it. Note that the two occurrences of u in s are redexes *created* by v_1 and v_2, as the occurrences of u in t and o are *not* λ_{LV}-redexes (they are not admissible). It is reasonable to share the occurrences of u in s, but in order to make the reduction graph of t a DFS (in particular, for the [contribution] axiom to be satisfied), we need to share v_1 and v_2 as well, although the two are not related.[4] This goes beyond Lévy's concept of sharing, and the resulting DFS is not separable as the contraction of w creates two different redexes, v_1 and v_2, of the same family.

[4] The redexes v_1 and v_2 get different labels in Levy's labelling system for the λ-calculus, and cannot be related either by zig-zag or by extraction.

Below, $FAM_z(P)$ denotes the set of zig-zag classes whose member redexes are contracted in P, in an ASDRS; and $Fam_z(Qu)$ denotes the zig-zag class of Qu. Further, if ϕ', ϕ are zig-zag classes, we write $\phi' \hookrightarrow_z \phi$ iff for any $Pu \in \phi$, P contracts a redex in ϕ'.

Lemma 45 *If $Pv \simeq_z Qw$, then $v/(Q/P) = w/(P/Q)$. In particular, where $P \approx_{STA} Q$, it follows that $v = w$.*

Proof *By Theorem 43, $Q \approx_L N + Q'$, $P \approx_L N + P'$, $w = u/Q'$ and $v = u/P'$, where Nu is a canonical form of Qw and Pv. Then $P/Q \approx_L P'/Q'$ and $Q/P \approx_L Q'/P'$. Hence $w/(P/Q) = w/(P'/Q') = u/(Q' \sqcup P') = u/(P' \sqcup Q') = v/(Q'/P') = v/(Q/P)$.*

Corollary 46 *Any reduction in an ASDRS $\mathcal{R}$ can contract at most one element of a zig-zag class.*

Proof *Let $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \ldots$ be a reduction in $\mathcal{R}$, and let $P_i : t_0 \twoheadrightarrow t_i$ be its initial part. If on the contrary $P_i u_i \simeq_z P_k u_k$ for some $i < k$, then we must have by Lemma 45 that $u_i/(u_i + \ldots + u_{k-1}) = u_k$ – a contradiction.*

As a consequence, we get that $\simeq_z$ satisfies [FFD] in any ASDRS. To show that $\simeq_z$ satisfies [creation] too, we need a few lemmas.

Lemma 47 *Let P be Q-needed, in an ASDRS. Then $FAM_z(P) \subseteq FAM_z(Q)$. In particular, if $P \in STV(Q)$, then $FAM_z(P) \subseteq FAM_z(Q)$, and if $P \approx_{STA} Q$, then $FAM_z(P) = FAM_z(Q)$.*

Proof *Let v be a contracted redex in P, say $P = P' + v + P''$. Then v is Q/P'-needed. Hence $Fam_z(P'v) \in FAM_z(Q/P') \subseteq FAM_z(Q)$, implying the lemma.*

Lemma 48 *Let $Q^* : t \overset{P}{\twoheadrightarrow} s \xrightarrow{u} e$ where u creates $v \in e$. Then, for any canonical form $Q'v'$ of Q^*v, Q' contracts a redex zig-zag related to Pu.*

Proof *We have by Lemma 38 that $Q = ST(Q^*) = P' + u'$, where $P \approx_L P' + P''$ (for some Q-unneeded P'') and $u = u'/P''$. If Qv is not a canonical form, by Lemma 38 there is an extraction step $Qv \xrightarrow{w_1} Q_1 v_1$ (i.e., $Q \approx_{STA} Q_1 + w_1$ and $v = v_1/w_1$). Since $Q_1 + w_1 \approx_{STA} Q = P' + u'$, we have by Lemma 41 that $Q_1 \approx_{STA} P_1 + u_1$ such that $P_1 u_1 \simeq_z P'u' \simeq_z Pu$. So we have $(P' + u')v \xrightarrow{w_1} (P_1 + u_1)v_1$ such that $P_1 u_1 \simeq_z P'u'$. Similarly, if $(P_1 + u_1)v_1$ is not a canonical form, there is an extraction step $(P_1 + u_1)v_1 \xrightarrow{w_2} (P_2 + u_2)v_2$ such that $P_2 u_2 \simeq_z P_1 u_1 \simeq_z P'u' \simeq_z Pu$, and so on. So a canonical form of Qv has the form $(P_m + u_m)v_m$ such that $Pu \simeq_z P_m u_m$. Since, by Theorem 42, for any canonical form $Q'v'$ of Qv (and hence of Q^*v), $Q' \approx_{STA} P_m + u_m$ and $v_m = v'$, it follows by Lemma 47 that Q' contracts a redex in the family of Pu.*

Lemma 49 *Let* $Pv \xrightharpoonup{w} P'v'$. *Then* $FAM_z(P') \subseteq FAM_z(P)$.

Proof *By Definition 39,* $Pv \xrightharpoonup{w} P'v'$ *implies that* $P' + w \in STV(P)$, *and by Lemma 47,* $FAM_z(P') \subseteq FAM_z(P' + w) \subseteq FAM_z(P)$.

The next lemma implies that $\simeq_z$ satisfies [creation], in any ASDRS.

Lemma 50 *Let* $Q : e \overset{P}{\twoheadrightarrow} t \xrightarrow{u} s$ *where* u *creates* $v \in s$. *Then* $Fam_z(Pu) \hookrightarrow_z Fam_z(Qv)$.

Proof *By Lemma 48, if* $Q'v'$ *is a canonical form of* Qv, *then* $Fam_z(Pu) \in FAM_z(Q')$. *Now it follows from Lemmas 47 and 49 and Theorem 43 that for any* $Q^*v^* \simeq_z Qv$, $Fam_z(Pu) \in FAM_z(Q^*)$, *i.e.,* $Fam_z(Pu) \hookrightarrow_z Fam_z(Qv)$.

Theorem 51 *Let* $\mathcal{R}$ *be a non-duplicating SDRS. Then* $\mathcal{F}_R = (R, \simeq_z, \hookrightarrow_z)$ *is a non-duplicating zig-zag DFS.*

Proof *We need to show that* $\mathcal{F}_R$ *satisfies all family axioms. [contribution] is immediate by the definition of* $\hookrightarrow_z$. *Since for any* $u, v \in t$, $\emptyset_t u$ *and* $\emptyset_t v$ *are canonical forms,* $u \neq v$ *implies by Theorem 43 that* $\emptyset_t u \not\simeq_z \emptyset_t v$, *i.e., [initial] holds. [creation] is immediate from Lemma 50, and [FFD] from Corollary 46.*

Next we show that an affine DFS is a zig-zag DFS iff it is separable. First we establish a characterization of separability of a DFS $\mathcal{F}$ via uniqueness of contracted families in reductions in $\mathcal{F}$. It shows that, in separable DFSs, and only in such DFSs, there is no sharing (in the affine case) – all reductions are in fact complete family-reductions.

Lemma 52 *Let* $\mathcal{F} = (\mathcal{R}, \simeq, \hookrightarrow)$ *be an affine DFS. Then the following are equivalent:*

(1) $\mathcal{F}$ *is separable;*
(2) *Elements of any family are contracted at most once in any reduction in* $\mathcal{F}$*;*
(3) $Pv \simeq Pv'$ *implies* $v = v'$*;*
(4) *Any reduction in* $\mathcal{F}$ *is in fact a complete family-reduction, and vice versa.*

Proof

(1) $\Rightarrow$ (2)*: Let* $P : t_0 \overset{P_i}{\twoheadrightarrow} t_i \xrightarrow{u_i} t_{i+1} \twoheadrightarrow t_n$, *and let us prove that* $k < m$ *implies that* $P_k u_k \not\simeq P_m u_m$ *by induction on the number of families contributing to* $Fam(P_k u_k)$. *Suppose on the contrary that* $\phi = Fam(P_k u_k) = Fam(P_m u_m)$. *Let* $P'_k u'_k$ *be a canonical form of* $P_k u_k$, *which exists by Theorem 42. Hence there is* Q'_k *such that* $P'_k + Q'_k \approx_L P_k$, *and* $u_k = u'_k / Q'_k$. *So we have that* $P_{k+1} = P_k + u_k \approx_L P'_k + Q'_k + u_k \approx_L P'_k + u'_k + Q'_k / u'_k$ *(see the figure below). Since by [contribution]* P'_k *contracts redexes in all contributor families of* $Fam(P'_k u'_k) = \phi$, *and since by the induction assumption no redexes in these*

families can be contracted again, u_m is not created by its preceding step in $u'_k + Q'_k/u'_k + u_{k+1} + \ldots + u_{m-1}$, by [creation]. Similarly, its ancestor redex is not a created redex, and so on. That is, u_m is a residual of some redex u'_m in the final term of P'_k, different from u'_k. Hence $Fam(P'_k u'_k) = \phi = Fam(P'_k u'_m)$. Since $P'_k u'_k$ is in extraction normal form, the last step of P'_k, denote it by $P''_k u''_k$, creates u'_k. Hence $\psi = Fam(P''_k u''_k) \hookrightarrow \phi$ by [creation]. Since by the induction assumption P''_k does not contract redexes of the same family more than once, and by [contribution] the history of any member of ϕ must contract a redex in ψ, u''_k must create u'_m too, contradicting separability.

$$\begin{tikzcd}
\cdot \arrow[r, two heads, "P'_k"] & \cdot \arrow[r, two heads, "Q'_k"] \arrow[d, "u'_k"'] & \cdot \arrow[d, "u_k"] & & & \\
& \cdot \arrow[r, two heads, "Q'_k/u'_k"'] & \cdot \arrow[r, "u_{k+1}"'] & \cdot \arrow[r, two heads] & \cdot \arrow[r, "u_{m-1}"'] & \cdot
\end{tikzcd}$$

(2) $\Rightarrow$ (3)*: If there were $Pv \simeq Pv'$ with $v \neq v'$, then at least one of $P + v + v'/v$, $P + v' + v/v'$ would contract two members of $Fam(Pv)$ by [weak acyclicity], contradicting (2).*

(3) $\Rightarrow$ (4)*: Immediate.*

(4) $\Rightarrow$ (1)*: If $\mathcal{F}$ was not separable, then there would be Pv, w' and w'' such that v creates both w' and w'', $w' \neq w''$, and $Fam((P + v)w') = Fam((P + v)w'')$. By the assumption (4), the reduction $P + v$ is also a complete family-reduction, implying that $P + v + w' \| w''$, where $w' \| w''$ is the multi-step contracting w' and w'' in parallel, is a complete family-reduction which is not a reduction, contradicting (4).*

Lemma 53 *Let Pv be a canonical element of a family ϕ, in an affine DFS $\mathcal{F} = (R, \simeq, \hookrightarrow)$, and let P contract a redex w. Then $Fam(w) \hookrightarrow \phi$.*

Proof *Suppose on the contrary that $\psi = Fam(w) \not\hookrightarrow \phi$, and assume that Pv and w are such that w is (one of) the latest among steps in canonical elements of ϕ that do not contribute to ϕ. Since Pv is canonical, w cannot be the last step of P as the last step of P creates v by Definition 37, and therefore its family contributes to ϕ by [creation]. Further, if v is the next to w step in P, then w cannot create v as this would imply $\psi \hookrightarrow Fam(v)$ by [creation], implying $\psi \hookrightarrow \phi$ (as $Fam(v) \hookrightarrow \phi$ by the choice of w). Hence w can be permuted with v in P, yielding again (by Lemma 33) a canonical element of ϕ in which a step whose family does not contribute to ϕ is contracted later than w in P – a contradiction.*

Lemma 54 *Let P and Q be standard co-initial finite reductions such that $FAM_z(P) = FAM_z(Q)$. Then $P \approx_L Q$.*

Proof *Suppose on the contrary that $P \not\approx_L Q$, and say $P/Q \neq \emptyset$. Then P contracts a redex u, say $P = P' + u + P''$, such that $u/(Q/P') \neq \emptyset$. Let v be a step in Q, i.e., $Q = Q' + v + Q''$ (see the figure). Then if $u' = u/(Q'/P')$*

and $v' = v/(P'/Q')$, we have $u' \neq v'$, thus $(P' \sqcup Q')u' \neq (P' \sqcup Q')v'$. Hence, by Lemma 45, $Fam_z(P'u) = Fam_z((P' \sqcup Q')u') \neq Fam_z((P' \sqcup Q')v') = Fam_z(Q'v)$, i.e., $FAM_z(P) \ni Fam_z(P'u) \notin FAM_z(Q)$ – contradiction.

$$
\begin{tikzcd}
\cdot \arrow[r, two heads, "P'"] \arrow[d, two heads, "Q'"'] & \cdot \arrow[r, "u"] \arrow[d, two heads] & \cdot \arrow[r, two heads, "P''"] \arrow[d, two heads] & \cdot \arrow[d, two heads, "\emptyset"] \\
\cdot \arrow[r, two heads] \arrow[d, "v"'] & \cdot \arrow[r, "u'"] \arrow[d, "v'"] & \cdot \arrow[r, two heads] \arrow[d, "\emptyset"] & \cdot \arrow[d, "\emptyset"] \\
\cdot \arrow[r, two heads] \arrow[d, two heads, "Q''"'] & \cdot \arrow[r, two heads, "\emptyset"'] \arrow[d, two heads] & \cdot \arrow[r, two heads] \arrow[d, two heads] & \cdot \arrow[d, two heads, "\emptyset"] \\
\cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot \arrow[r, two heads, "\emptyset"'] & \cdot
\end{tikzcd}
$$

Theorem 55 *An affine DFS $\mathcal{F}$ is separable iff it is a zig-zag DFS.*

Proof

(⇒) Let $\mathcal{F} = (\mathcal{R}, \simeq_s, \hookrightarrow_s)$ be separable, and suppose on the contrary that $\simeq_s \neq \simeq_z$. Then, by $\simeq_z \subseteq \simeq_s$, there is a $\simeq_s$-family ϕ which contains at least two zig-zag classes ϕ' and ϕ'', and we can assume that ϕ has a minimal number of $\hookrightarrow_s$-contributor $\simeq_s$-families $\phi_1, \ldots, \phi_n$. By the minimality of ϕ, $\phi_1, \ldots, \phi_n$ are also $\simeq_z$-families. Let $P'v'$ and $P''v''$ be canonical elements of ϕ' and ϕ''. By Lemma 53 and [contribution], P' and P'' both contract exactly the elements of (all) families $\phi_1, \ldots, \phi_n$, thus $FAM_z(P') = FAM_z(P'')$, implying by Lemma 54 that $P' \approx_{STA} P''$. Hence the last step of P' creates two different members $P'v'$ and $P'v''$ of a $\simeq_s$-family ϕ – a contradiction.
(⇐) Immediate from Lemma 52 and Corollary 46.

We conclude this section by a useful characterization of histories of canonical elements of zig-zag (hence extraction and separable) families, in ASDRSs. In particular, it implies that the history of a canonical redex is indeed a shortest reduction creating a member of the corresponding family.

Theorem 56 *Let Pv be a canonical element of a family ϕ, in an AZDFS $\mathcal{F}_R = (R, \simeq_z, \hookrightarrow_z)$. Then P contracts exactly one redex in every contributor family of ϕ.*

Proof *By Lemma 53 and [contribution], P contracts exactly redexes in (all) contributor families of ϕ. The uniqueness of contracted families follows from Lemma 52 and Theorem 55.*

7 Implementation DFSs

We now define the *implementation* $\mathcal{F}_I$ of a DFS $\mathcal{F}$, whose reduction steps correspond to complete family-reductions in $\mathcal{F}$, hence the name. We also show that optimal reductions in $\mathcal{F}$, relative to any stable set $\mathcal{S}$ of results, are implemented in $\mathcal{F}_I$ by the shortest $\mathcal{S}_I$-normalizing reductions, where $\mathcal{S}_I$ is the set of terms in $\mathcal{F}_I$ corresponding to terms in $\mathcal{S}$. We will assume that the reduction graph of $\mathcal{F}$ is the reduction graph of an *initial* term, denoted by $t_\emptyset$, and that families are considered relative to $t_\emptyset$, i.e, all histories start with $t_\emptyset$.

Definition 57 *Let $\mathcal{F} = (\mathcal{R}, \simeq, \hookrightarrow)$ be a DFS where $\mathcal{R} = (A, /)$ and $A = (Ter, Red, \rightarrow)$. Then the* implementation *or* Lévy-implementation *of $\mathcal{F}$ is the AZDFS $\mathcal{F}_I = (\mathcal{R}_I, \simeq_I, \hookrightarrow_I)$, where*

- *the branches of the reduction graph of the underlying ARS A_I of $\mathcal{R}_I = (A_I, /_I)$ are complete family-reductions starting from $t_\emptyset$, each edge (i.e., reduction step) being a multi-step contracting a family of redexes.*
- *the residual relation $/_I$ is defined as follows: let U and V be complete sets of redexes in two $\simeq$-families, in a term s, and let $U : s \rightarrow_I o$ be the multi-step contracting U. Then $V/_IU$ is the multi-step $o \rightarrow_I e$ contracting all members of the set V/U.*
- *the family and contribution relations $\simeq_I, \hookrightarrow_I$ in $\mathcal{F}_I$ are those induced by $\simeq$ and $\hookrightarrow$: let P_I and Q_I be reductions in $\mathcal{R}_I$ corresponding to complete family-reductions P and Q in $\mathcal{F}$; then $P_IU \simeq_I Q_IV$ iff for any $u \in U, v \in V$, $Pu \simeq Qv$; and $Fam(P_IU) \hookrightarrow_I Fam(Q_IV)$ iff $Fam(Pu) \hookrightarrow Fam(Qv)$.*

We need to verify that $\mathcal{F}_I$ in the above definition is indeed an AZDFS.

Lemma 58 *Let $P : t_\emptyset \twoheadrightarrow s$ be a complete family-reduction in a DFS $\mathcal{F} = (R, \simeq, \hookrightarrow)$, let $U, V \subseteq s$ be complete sets of redexes of families ϕ and ψ in s, respectively, and let $s \overset{V}{\twoheadrightarrow} o$. Then $U' = U/V$ is the complete set of redexes of ϕ in o.*

Proof *Since $\simeq_z \subseteq \simeq$, U' consists of redexes of ϕ. Suppose on the contrary that there is $w \in o$ such that $(P+V)w \in \phi$ and $w \notin U'$. Again by $\simeq_z \subseteq \simeq$, w must be created along V, and we have by [creation] that $\psi \hookrightarrow \phi$. But this implies by [contribution] that P contracts a member of ψ, and therefore the complete family ψ (since P is a complete family-reduction), and $P + V$ contracts the family ψ twice, contradicting Lemma 30.*

Thus the residual relation is well defined. Obviously, $V/_IV = \emptyset$, and every family in o has at most one ancestor family in s. Further, [weak acyclicity] and [stability] for $\mathcal{F}_I$ follow immediately from the Weak Acyclicity Lemma and the Stability Lemma, respectively (one just needs to take for the reductions P and Q in these lemmas complete developments of disjoint sets of redexes, which are clearly external). The axiom [initial] in $\mathcal{F}_I$ follows immediately from [initial] in $\mathcal{F}$. If $P+U+V$ is a reduction in $\mathcal{F}_I$ such that U creates V, then the redexes in V are created along U (when $P+U+V$ is considered as a reduction in $\mathcal{F}$), i.e., $Fam(U) \hookrightarrow Fam(V)$ in $\mathcal{F}$, hence $Fam(U) \hookrightarrow_I Fam(V)$, implying [creation] in $\mathcal{F}_I$. Since complete family-reductions can be viewed as reductions in $\mathcal{F}$ (by considering multi-steps as corresponding complete developments), [contribution] for $\hookrightarrow_I$ follows immediately from [contribution] for $\hookrightarrow$. Finally, [FFD] for $\mathcal{F}_I$ follows immediately from Lemma 30 for $\mathcal{F}$. Hence $\mathcal{F}_I$ is indeed a DFS as $\simeq_I$ clearly contains the zig-zag relation. Note that $\mathcal{F}_I$ is separable as its steps contract entire $\simeq$-families, hence it is an AZDFS by Theorem 55:

Theorem 59 *For any DFS $\mathcal{F}$, $\mathcal{F}_I$ is an AZDFS.*

Next we show that any sharing $\simeq$ in an SDRS stronger than zig-zag can be decomposed into any weaker sharing $\simeq'$ and a *non-separable* sharing $\simeq^*$ in the implementation of $\simeq'$.

Definition 60 *Let $\mathcal{F} = (\mathcal{R}, \simeq)$ and $\mathcal{F}' = (\mathcal{R}, \simeq')$ be DFSs. We say that $\mathcal{F}$ has a* stronger *sharing than $\mathcal{F}'$, written $\mathcal{F} \geq \mathcal{F}'$, if $\simeq' \subseteq \simeq$.*

Theorem 61 *Let $\mathcal{F} = (\mathcal{R}, \simeq)$ and $\mathcal{F}' = (\mathcal{R}, \simeq')$ be DFSs. Then $\mathcal{F} > \mathcal{F}'$ iff there is a non-separable family relation $\simeq^*$ on the implementation DRS $\mathcal{R}'_I$ of $\mathcal{F}'$ such that $\mathcal{F}^*_I = \mathcal{F}_I$, where $\mathcal{F}^* = (\mathcal{R}'_I, \simeq^*)$.*

Proof

($\Rightarrow$) Let $\mathcal{F} > \mathcal{F}'$. Define $\simeq^*$ on $\mathcal{R}'_I$ by: $Pv \simeq^* Qu$ iff $P'v' \simeq Q'u'$, where P' and Q' are reductions in $\mathcal{R}$ corresponding to P and Q, and v' and u' are $\mathcal{R}$-redexes in redex-sets contracted in multi-steps v and u. It is immediate that the definition is correct (since reductions in $\mathcal{F}$ corresponding to a reduction in $\mathcal{R}'_I$ are all sequentializations of a multi-step reduction and are Lévy-equivalent, and since $\mathcal{F} > \mathcal{F}'$). Further, the family axioms for $\simeq^*$ follow from those of $\mathcal{F}$ exactly as they were verified above for $\mathcal{F}_I$ in the place of $\mathcal{F}^*$ (the only difference is that $\mathcal{F}^*$ is a 'partial implementation' of $\mathcal{F}$ while $\mathcal{F}_I$ is the 'complete' or Lévy-implementation). By the definition of $\simeq^*$, complete family-reductions in $\mathcal{F}^*$ are exactly complete family-reductions in $\mathcal{F}$, and $\mathcal{F}^*_I = \mathcal{F}_I$ follows since both are AZDFSs by Theorem 59. Since $\mathcal{F} > \mathcal{F}'$ and $\mathcal{F}'_I$ is an AZDFS, $\simeq^*$ is strictly larger than zig-zag, hence is non-separable by Theorem 55.

($\Leftarrow$) Immediate from Definition 60.

Thus, in particular, the study of a sharing in an SDRS strictly larger than the zig-zag can be reduced to studying zig-zag (when it is a family-relation) and studying non-separable affine families.

The following lemmas relate neededness in a DFSs with neededness in its implementation. Together with Theorem 31, they imply that the implementation DRSs $\mathcal{F}_I$ do indeed correctly implement complete family-reductions in DFSs $\mathcal{F}$: If $\mathcal{S}_I$ is the set of terms in $\mathcal{F}_I$ corresponding to terms in a stable set $\mathcal{S}$ in $\mathcal{F}$, then $\mathcal{S}_I$ is stable and $\mathcal{S}_I$-needed reductions in $\mathcal{F}_I$ are actually $\mathcal{S}_I$-needed complete family-reductions by Lemma 52.(4) and Theorem 59. Hence $\mathcal{S}_I$-needed reductions are optimal w.r.t. $\mathcal{S}_I$ and implement optimal family-reductions in $\mathcal{F}$, w.r.t $\mathcal{S}$.

Lemma 62 *Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$, and let $\mathcal{S}_I = \mathcal{S} \cap Ter(\mathcal{F}_I)$, where $Ter(\mathcal{F}_I)$ is the set of terms in $\mathcal{F}$. Then $\mathcal{S}_I$ is stable in $\mathcal{F}_I$.*

Proof *Let $t \xrightarrow{u}_I s$ in $\mathcal{F}_I$, where $t \notin \mathcal{S}_I$ and $s \in \mathcal{S}_I$. Then $t \stackrel{U}{\twoheadrightarrow} s$, where $t \notin \mathcal{S}$ and $s \in \mathcal{S}$. By closure of $\mathcal{S}$ under unneeded expansion, any sequentialization of $t \stackrel{U}{\twoheadrightarrow} s$ must contract at least one $\mathcal{S}$-needed redex, thus by Lemma 8, $U \subseteq t$ must contain at least one $\mathcal{S}$-needed redex. Hence there is no $\mathcal{S}$-normalizing reduction in $\mathcal{F}$ external to U, thus there is no $\mathcal{S}_I$-normalizing reduction in $\mathcal{F}_I$ external to u, i.e., u is $\mathcal{S}_I$-needed. Closure of $\mathcal{S}_I$ under reduction follows from that for $\mathcal{S}$, and we are done.*

Lemma 63 *Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$ not containing the initial term $t_0 = t_\emptyset$, let $P : t_0 \stackrel{U_0}{\twoheadrightarrow} t_1 \stackrel{U_1}{\twoheadrightarrow} \ldots \twoheadrightarrow t_n$ be an $\mathcal{S}$-normalizing complete family-reduction in $\mathcal{F}$, and let $P_I : t_0 \xrightarrow{u_0}_I t_1 \xrightarrow{u_1}_I \ldots \to_I t_n$ be its corresponding reduction in $\mathcal{F}_I$. Then P is $\mathcal{S}$-needed iff P_I is $\mathcal{S}_I$-needed.*

Proof

($\Rightarrow$) Assume that P be $\mathcal{S}$-needed, and suppose on the contrary that P_I is not $\mathcal{S}_I$-needed, i.e., u_i is $\mathcal{S}_I$-unneeded for some i. Then there is an $\mathcal{S}_I$-normalizing reduction $N : t_i \twoheadrightarrow_I s$ in $\mathcal{F}_I$ that is external to u_i. It remains to show that any corresponding complete family-reduction N' in $\mathcal{F}$ (no matter how the multi-steps are sequentialized) is external to U_i: the latter implies that U_i does not contain an $\mathcal{S}$-needed redex, contradicting $\mathcal{S}$-neededness of P. If on the contrary a multi-step W of N' contracts a residual of a redex in U_i, then it follows from Lemma 58 that W is the residual of U_i along N' (as both U_i and W are complete sets of redexes of the same family in corresponding terms), implying that the corresponding step of N is a residual of u_i – a contradiction.

($\Leftarrow$) Let P_I be $\mathcal{S}_I$-needed. Suppose on the contrary that U_i is not $\mathcal{S}$-needed for some i. Let Q be an $\mathcal{S}$-needed $\mathcal{S}$-normalizing complete family-reduction (which exists by Corollary 29). Note that the residual of U_i in any term along Q forms a complete family by Lemma 58, and all residuals of redexes in U_i remain $\mathcal{S}$-unneeded by Lemma 8. Thus residuals of U_i along Q are redex-sets disjoint from the contracted redex-sets, implying that the corresponding reduction of Q in $\mathcal{F}_I$ is external to u_i, which contradicts $\mathcal{S}_I$-neededness of u_i.

Corollary 64 (Optimal Implementation) *Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$ and $\mathcal{S}_I$ be its corresponding stable set of terms in $\mathcal{F}_I$. Then $\mathcal{S}_I$-needed $\mathcal{S}_I$-normalizing reductions in $\mathcal{F}_I$ are $\mathcal{S}_I$-optimal in $\mathcal{F}_I$, and they implement $\mathcal{S}$-optimal complete family-reductions in $\mathcal{F}$.*

8 Conclusions and future work

We have introduced Deterministic Residual and family Structures and have proven two abstract versions of the RN theorem: one in stable DRSs for regular stable sets $\mathcal{S}$, and another in DFSs for all stable $\mathcal{S}$. We believe that our first proof is the simplest existing proof among those using the residual notion, though it covers all the existing normalization results, except for the one in [18,22], which is covered by our second RN theorem. It is remarkable that, unlike the proofs in [16,29,10], our first proof does not use the notion of standard reduction. Similar proofs for orthogonal CRSs in [32] and for orthogonal DAGs in [52,53] use an even stronger termination argument, expressed by the [FFD] axiom; they used suitable labelling systems to define notions of family. Our second proof can be seen as a generalization of that proof method, which was used earlier by Lévy in [50,51]. It would be interesting to investigate whether it is possible to prove our second theorem in the context of stable DRS, without invoking family axioms, but possibly using some much weaker axioms.

Further, we have introduced and studied an abstract concept of optimal implementation of DFSs, and showed that needed computations (w.r.t. stable sets of results) in implementation DRSs mimic optimal (in the sense of Lévy and beyond) computations in original DFSs. We have shown that every affine SDRS can be turned into an affine DFS by taking zig-zag as the family relation, and that zig-zag is the only family relation with the separability property – no redex can create two different members of the same family simultaneously. Finally, we have shown that sharing is compositional. In particular, any family relation can be decomposed into zig-zag (when it is a family relation) and a non-separable affine family-relation, which facilitates the study of complicated (non-separable) concepts of sharing in duplicating systems (such as the one in [4]). As for zig-zag in duplicating SDRSs, we believe that suitable extensions of SDRSs can be defined, in which one can extend our construction of zig-zag DFSs to the general (duplicating) case, and our proofs can be used as a basis. In the general case, the [FFD] axiom of DFSs becomes a strong requirement, and extra axioms will be necessary to ensure it. We expect that a modification of our extraction algorithm will be applicable.

Besides their clear relevance for the study of syntactic properties of orthogonal rewrite systems (which we hope to have amply demonstrated), DFSs have already proven their usefulness also in the study of concurrent semantics of orthogonal rewrite systems: Indeed, based on the results obtained in this paper, we have defined in [44] a fully adequate *Prime Event Structure* [70,58,71] style semantics for orthogonal rewrite systems in a uniform fashion. Furthermore in [42,43] we studied the concepts of *independence* and *interaction* of (sub)computations in the framework of DFSs, yielding a nice theory of *Euclidian Geometry* for reduction spaces in orthogonal rewrite systems. And we have also developed a relativized stable computational semantics for orthogonal reduction systems [38], extending the computational semantics of rewrite systems proposed by Boudol [13]. Among many possible directions for future work, let us mention that it is important to extend these results, both the syntactic and semantic aspects, to non-deterministic systems, enabling one to model languages for concurrency for example. Related work in this direction (not restricted to conflict-free systems) includes [13,15,65,54–56].

References

[1] S. Antoy, R. Echahed and M. Hanus, A needed narrowing strategy, in: *Proc. 21st ACM Symp. on Principles of Programming Languages* (1994) 268-279.

[2] S. Antoy and A. Middeldorp, A Sequential Reduction Strategy, *Theoret. Comput. Sci.* **165(1)** (1996) 75-95.

[3] Z.M. Ariola, M. Felleisen, J. Maraist, M. Odersky, P.A. Wadler, A call-by-need lambda calculus, in: *Proc. 22nd ACM Symp. on Principles of Programming Languages* (1995) 233-246.

[4] A. Asperti and C. Laneve, Interaction systems I: The theory of optimal reductions, *Mathematical Structures in Computer Science* **11** (1993) 1-48.

[5] A. Asperti and C. Laneve, Interaction systems II: The practice of optimal reductions, *Theoret. Comput. Sci.* **159(2)** (1996) 191-244.

[6] A. Asperti and H. Mairson, Parallel beta reduction is not elementary recursive, in: *Proc. 25th ACM Symp. on Principles of Programming Languages* (1998).

[7] A. Asperti and S. Guerrini, *The Optimal Implementation of Functional Programming Languages* (Cambridge Tracts in Theoretical Computer Science, Cambridge University Press, 199?). (To appear.)

[8] H. P. Barendregt, *The Lambda Calculus, Its Syntax and Semantics* (North-Holland, Amsterdam, 1984).

[9] H.P. Barendregt, J.A. Bergstra, J.W. Klop and H. Volken, Some notes on lambda-reduction, in: Degrees, reductions, and representability in the lambda calculus, Preprint no. 22, University of Utrecht, Department of mathematics, 1976, 13-53.

[10] H.P. Barendregt, J.R. Kennaway, J.W. Klop, M.R. Sleep, Needed reduction and spine strategies for the lambda calculus, *Information and Computation* **75(3)** (1987) 191-231.

[11] G. Berry, Modéles complétement adéquats et stables des λ-calculs typés, Thèse de l'Université de Paris VII, 1979.

[12] G. Berry and J.-J. Lévy, Minimal and optimal computations of recursive programs, *J. of the Association for Computing Machinery* **26** (1979) 148-175.

[13] G. Boudol, Computational semantics of term rewriting systems. in: M. Nivat and J.C. Reynolds, eds., *Algebraic methods in semantics* (Cambridge University Press, 1985) 169-236.

[14] A. Church and J.B. Rosser, Some properties of conversion, *Transactions of American Mathematical Society* **39** (1936) 472-482.

[15] D. Clark and J.R. Kennaway, Event structures and non-orthogonal term graph rewriting, *Mathematical Structures in Computer Science* **6** (1996) 545-578.

[16] H.B. Curry and R. Feys, *Combinatory Logic, Vol. 1* (North-Holland, Amsterdam, 1958).

[17] P. Gardner, Discovering needed reductions using type theory, in: *Proc. TACS'94*, Lecture Notes in Computer Science, vol 789 (Springer, Berlin, 1994) 555-574.

[18] J.R.W. Glauert and Z. Khasidashvili, Relative normalization in orthogonal expression reduction systems, in: *Proc. CTRS'94*, Lecture Notes in Computer Science, vol. 968 (Springer, Berlin, 1994) 144-165.

[19] J.R.W. Glauert and Z. Khasidashvili, Minimal and optimal relative normalization in orthogonal expression reduction systems, Technical Report SYS-C94-06, UEA Norwich, 1994.

[20] J.R.W. Glauert and Z. Khasidashvili, Relative normalization in deterministic residual structures, in: *Proc. CAAP'96*, Lecture Notes in Computer Science, vol. 1059 (Springer, Berlin, 1996) 180-195.

[21] J.R.W. Glauert and Z. Khasidashvili, Relative normalization in stable deterministic residual structures, Technical Report SYS-C96-05, UEA Norwich, 1996.

[22] J.R.W. Glauert and Z. Khasidashvili, Stable results and relative normalization. (Submitted.)

[23] G. Gonthier, M. Abadi and J.-J. Lévy, The geometry of optimal lambda reduction, in: *Proc. 19th ACM Symp. on Principles of Programming Languages* (1992) 15-26.

[24] G. Gonthier, J.-J. Lévy and P.-A. Melliès, An abstract standardisation theorem, in: *Proc. 7th IEEE Symp. on Logic in Computer Science* (1992) 72-81.

[25] S. Guerrini, Theoretical and practical issues of optimal implementations of functional languages, Ph.D. Thesis, University of Pisa, 1996.

[26] R.J. Hindley, An abstract form of the Church-Rosser theorem I, *Journal of Symbolic Logic* **34(4)** (1969) 545-560.

[27] B. Hoffmann and D. Plump, Implementing term rewriting by jungle evaluation, *Theoretical Informatics and Applications* **25** (1991) 445-472.

[28] G. Huet, Confluent reductions: Abstract properties and applications to term rewriting systems, *J. of the Association for Computing Machinery* **27(4)** (1980) 797-821

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

[30] V. Kathail, Optimal interpreters for lambda-calculus based functional languages, Ph.D. Thesis, MIT, 1990.

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

[32] J.R. Kennaway, M.R. Sleep, Neededness is hypernormalizing in regular combinatory reduction systems, Preprint, School of Information Systems, UEA Norwich, 1989.

[33] J.R. Kennaway, J.W. Klop, M.R. Sleep and F.-J. de Vries, Event structures and orthogonal term graph rewriting, in: M.R. Sleep, M.J. Plasmeijer, M.C.J.D. van Eekelen, eds., *Term Graph Rewriting: Theory and Practice* (John Wiley, 1993) 141-156.

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

[35] Z. Khasidashvili, β-reductions and β-developments of λ-terms with the least number of steps, in: *Proc. COLOG'88*, Lecture Notes in Computer Science, Vol. 417 (Springer, Berlin, 1990) 105–111.

[36] Z. Khasidashvili, The Church-Rosser theorem in orthogonal combinatory reduction systems, Report 1825, INRIA Rocquencourt, 1992.

[37] Z. Khasidashvili, Optimal normalization in orthogonal term rewriting systems, in: *Proc. RTA'93*, Lecture Notes in Computer Science, Vol. 690 (Springer, Berlin, 1993) 243-258.

[38] Z. Khasidashvili, Stable computational semantics of conflict-free rewrite systems. (Draft.)

[39] Z. Khasidashvili and J.R.W. Glauert, Discrete normalization and standardization in deterministic residual structures, in: *Proc. ALP'96*, Lecture Notes in Computer Science, vol. 1139 (Springer, Berlin, 1996) 135-149.

[40] Z. Khasidashvili and J.R.W. Glauert, Discrete normalization and standardization in stable deterministic residual structures, Technical Report SYS-C96-06, UEA Norwich, 1996.

[41] Z. Khasidashvili and J.R.W. Glauert, Zig-zag, extraction and separable families in non-duplicating stable deterministic residual structures, Technical Report IR-420, Free University, Amsterdam, 1997.

[42] Z. Khasidashvili and J.R.W. Glauert, The geometry of orthogonal reduction spaces, in: *Proc. ICALP'97*, Lecture Notes in Computer Science, vol. 1256 (Springer, Berlin, 1997) 649-659.

[43] Z. Khasidashvili and J.R.W. Glauert, The geometry of conflict-free reduction spaces, Technical Report SYS-C98-04, UEA Norwich, 1998.

[44] Z. Khasidashvili and J.R.W. Glauert, Relating conflict-free stable transition and event models (extended abstract), in: *Proc. MFCS'97*, Lecture Notes in Computer Science, vol. 1295 (Springer, Berlin, 1997) 269-278.

[45] Z. Khasidashvili and J.R.W. Glauert, An abstract concept of optimal implementation, Technical Report SYS-C97-01, UEA Norwich, 1997.

[46] J.W. Klop, *Combinatory reduction systems* (Mathematical Centre Tracts no. 127, Amsterdam, 1980).

[47] J.W. Klop, Term rewriting systems, in: S. Abramsky, D. Gabbay and T. Maibaum, eds., *Handbook of Logic in Computer Science, Vol. 2* (Oxford University Press, 1992) 1-116.

[48] Y. Lafont, Interaction nets, in: *Proc. 17th ACM Symp. on Principles of Programming Languages* (1990) 95-108.

[49] J. Lamping, An algorithm for optimal lambda calculus reduction, in: *Proc. 17th ACM Symp. on Principles of Programming Languages* (1990) 6-30.

[50] J.-J. Lévy, Réductions correctes et optimales dans le lambda-calcul, Thèse de l'Université de Paris VII, 1978.

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

[52] L. Maranget, Optimal derivations in weak λ-calculi and in orthogonal term rewriting systems, in: *Proc. 18th ACM Symp. on Principles of Programming Languages* (1991) 255-269.

[53] L. Maranget L. La stratégie paresseuse, Thèse de l'Université de Paris VII, 1992.

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

[55] P.-A. Melliès, Axiomatic rewriting theory II: The lambda-sigma-calculus has the finite cone property, submitted.

[56] P.-A. Melliès, Axiomatic rewriting theory IV: A stability theorem in rewriting theory, in: *Proc. 14th IEEE Symp. on Logic in Computer Science* (1998).

[57] A. Middeldorp, Call by need computations to root-stable form, in: *Proc. 24th ACM Symp. on Principles of Programming Languages* (1997) 94-105.

[58] M. Nielsen, G. Plotkin and G. Winskel, Petri nets, event structures and domains, Part 1, *Theoret. Comput. Sci.* **13** (1981) 85-108.

[59] E. Nöcker, Efficient functional programming: Compilation and programming techniques, Ph.D. Thesis, Katholic University of Nijmegen, 1994.

[60] M. O'Donnell, Reduction strategies in subtree replacement systems, Ph.D. thesis, Cornell University, Ithaca, N.Y., 1976.

[61] V. van Oostrom, Confluence for abstract and higher-order rewriting, Ph.D. Thesis, Free University, Amsterdam, 1994.

[62] V. Van Oostrom, Higher order families, in: *proc. RTA'96*, Lecture Notes in Computer Science, vol. 1103 (Springer, Berlin, 1996) 392-407.

[63] V. Van Oostrom, Finite family developments, in: *proc. RTA'97*, Lecture Notes in Computer Science, vol. 1232 (Springer, Berlin, 1997) 308-322.

[64] G. Plotkin, Call-by-name, call-by-value and the λ-calculus, *Theoret. Comput. Sci.* **1** (1975) 125-159.

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

[66] B.K. Rosen, Tree-manipulating systems and Church-Rosser theorems, *J. of the Association for Computing Machinery* **20** (1973) 160-187.

[67] R.C. Sekar and I.V. Ramakrishnan, Programming in equational logic: Beyond strong sequentiality, *Information and Computation* **104(1)** (1993) 78-109.

[68] E.W. Stark, Concurrent transition systems, *Theoret. Comput. Sci.* **64(3)** (1989) 221-269.

[69] P.C. Wadsworth, Semantics and pragmatics of the lambda-calculus, Ph.D. thesis, University of Oxford, 1971.

[70] G. Winskel, Events in computation, Ph.D. Thesis, Edinburgh University, 1980.

[71] G. Winskel, An introduction to event structures, in: *Proc. Linear time, branching time and partial order in logics and models of concurrency*, Lecture Notes in Computer Science, vol. 354 (Springer, Berlin, 1989) 364-397.

[72] H. Xi, Evaluation under lambda abstraction, in: *proc. PLILP'97*, Lecture Notes in Computer Science, vol. 1292 (Springer, Berlin, 1997) 259-274.

[73] N. Yoshida, Optimal reduction in weak λ-calculus with shared environments, in: *Proc. ACM Conference on Functional Programming Languages and Computer Architecture* (1993) 243-252.