An Abstract Concept of Optimal Implementation

Zurab Khasidashvili [1]

Logic and Validation Technology
Intel, IDC, Haifa, Israel

John Glauert [2]

School of Computing Sciences, UEA
Norwich NR4 7TJ, UK

Abstract

In previous works, we introduced *Stable Deterministic Residual Structures* (SDRSs), Abstract Reduction Systems with an axiomatized residual relation which model orthogonal term and graph rewriting systems, and *Deterministic Family Structures* (DFSs), which add an axiomatized notion of *redex-family* to capture known *sharing* concepts in the λ-calculus and other orthogonal rewrite systems. In this paper, we introduce and study a concept of *implementation* of DFSs. We show that for any DFS $\mathcal{F}$, its implementation $\mathcal{F}_I$ is a non-duplicating DFSs 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. Further, we show that sharing is compositional: the sharing in a DFS $\mathcal{F}$ can be decomposed into a weaker sharing $\mathcal{F}'$ (such as zig-zag) and a sharing in the implementation $\mathcal{F}'_I$ of $\mathcal{F}'$ stronger than zig-zag. These results require 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.

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 [Lév78,Lév80]. Such an implementation has indeed been achieved by

[1] Email: Zurab.Khasidashvili@intel.com
[2] Email: J.Glauert@cmp.uea.ac.uk

Lamping and Kathail, reviving interest in optimal graph reduction. Since then, there have been many interesting discoveries around the theory and practice of optimality, and we refer the reader to [AG98] for a detailed treatment of the subject.

Lévy introduced the family concept in three different ways: via a suitable 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 Higher-Order Rewriting Systems (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 [Oos96]. However, the zig-zag family can be defined directly in R, and this yields a different, weaker, family concept [AL93]. Klop [Klo80], Maranget [Mar91], and others have defined similar yet different labellings for orthogonal (first or higher order) rewrite systems, all yielding a concept of family for these systems.

This variety of family concepts, and development of alternative graph rewriting algorithms for optimal implementation of orthogonal rewriting systems, such as Term Graph Rewriting [KKSV93], Jungle Rewriting [HP91], and many others, inspired by Wadsworth's original work on graph-based implementation of the λ-calculus [Wad71], created the need to develop an abstract notion of family general enough to cover all the existing notions, and refined enough to enable proof of normalization and optimality results. Such structures were introduced by the authors in [GK96] as *Deterministic Family Structures* (DFSs) building on recent developments of abstract reduction systems with an axiomatized residual relation, such as the *Concurrent Transition Systems* (CTSs) of Stark [Sta89] and the *Abstract Reduction Systems* (ARSs) of Gonthier et al [GLM92].

Our DFSs are defined as *Deterministic Residual Structures* (DRSs) with an axiomatized family relation. DRSs, in turn, are Abstract Reduction Systems with an axiomatized residual relation. Despite its highly abstract nature, a counterpart of Berry's *stability* property [Ber79] enabled us to prove the normalization theorem for all DRSs, and not only w.r.t. normal forms, but in general for *regular stable sets* of 'results', such as head-normal forms or Böhm tress [GK96,GK02].

In this paper, we continue the abstract study of optimality theory started in [GK96]. We introduce an abstract concept of *Lévy-implementation*: With a DFS $\mathcal{F}$ we associate a *non-duplicating* DRS $\mathcal{R}_I$, called the implementation of $\mathcal{F}$, whose steps exactly correspond to complete family-reduction steps in $\mathcal{F}$, thus, for example, they model sharing-graph implementation of Lévy's complete family-reductions in the λ-calculus. It is not difficult to show that needed reductions in $\mathcal{R}_I$ (w.r.t. any stable set of results) correspond exactly to needed complete family-reductions in $\mathcal{F}$, implying that the former indeed implement optimal computations in $\mathcal{F}$ in the sense of Lévy [Lév78,Lév80].

Further, we show that the family relation on $\mathcal{R}_I$ induced by that of $\mathcal{F}$ coincides with zig-zag, which is the weakest family sharing. At first sight, one might think that there is no need or possibility of a stronger sharing in non-duplicating SDRSs. However, this is not the case as the example of Asperti and Laneve [AL93] demonstrating 'inadequacy' of Lévy's extraction algorithm for Interaction Systems is, when considered as an Abstract Reduction System, a non-duplicating SDRS. This example also shows that zig-zag does not coincide with the labelling and their extraction families based on the *shift* operation. These effects are caused by the fact that more than one members of a family can be created by a single step – the reduction step $t = \mu(\lambda x.(xx)) \to (\mu(\lambda x.(xx))\,\mu(\lambda x.(xx))) = s$, according to the μ-rule $\mu(\lambda x.X) \to [\mu(\lambda x.X)/x]X$, simultaneously creates the two μ-redexes in s; these redexes are intuitively in the same family, but cannot be related by zig-zag, nor by an extraction procedure similar to Lévy's. We call such families *non-separable.*

Actually, we show that the sharing concept formalized in DFSs is *compositional*: any sharing can be decomposed into a weaker sharing (such as zig-zag, when the latter forms a family-relation) 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.

Section 3 gives a characterization of zig-zag relation via extraction, used in Section 4 to prove that zig-zag is a family relation in every non-duplicating stable DRS. In section 5 we show that a family relation in a non-duplicating SDRS is separable iff it is zig-zag. In Section 6 we define and study implementation DRSs, and show the compositionality of sharing. Conclusions appear in Section 7. Omitted proofs can be found in Appendices B and C.

2 Affine DFSs

We start by introducing some notation.

Definition 2.1 We define an ARS as a triple $A = (\mathit{Ter}, \mathit{Red}, \to)$ where *Ter* is a set of *terms*,[3] ranged over by t, s, o, e; *Red* is a set of *redexes* (or *redex occurrences*), ranged over by u, v, w; and $\to: \mathit{Red} \to (\mathit{Ter} \times \mathit{Ter})$ is a function such that for any $t \in \mathit{Ter}$ there is only a finite set of $u \in \mathit{Red}$ such that $\to(u) = (t, s)$, 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 $\to$ is a *total* function, so one can identify u with the triple $t \xrightarrow{u} s$. A *reduction* is a sequence $t \xrightarrow{u_1} t_2 \xrightarrow{u_2} \ldots$. Reductions are denoted by P, Q, N. We write $P : t \twoheadrightarrow s$ or $t \overset{P}{\twoheadrightarrow} s$ if P denotes a reduction (sequence) from t to s. $|P|$ denotes the length of P. $P + Q$ denotes the concatenation of P and Q. We use U, V, W to denote sets

[3] We use the term 'term' rather than say 'object', but note that our terms may model various objects, such as graphs.

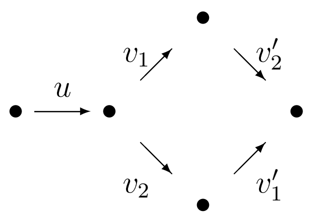
of redexes of a term.

Since there has been a series of publications using the framework of SDRSs and DFSs [GK96,KG97a,GK02,KG02], we do not recall SDRSs and DFSs in the main body of the paper. The following special kinds of DFSs will play an important role in this paper.

Definition 2.2 (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



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 (without the name) are studied in [AL93] for Interaction Systems, where it is demonstrated that such a family relation is in general strictly larger than the zig-zag. Another important example where a non-separable sharing is reasonable is the *lazy call-by-value* λ-calculus, λ_{LV} [Plo75]. It 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

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

where u is a λ_{LV}-redex such that x does not occur free in it. It is reasonable to share the two 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. 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. Note that the occurrences of u in s are *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).

3 Equivalence of Zig-zag and Extraction

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.

Lévy introduced an extraction procedure for the λ-calculus in [Lév78,Lév80] in order to prove decidability of the family relation. His extraction procedure is effective, and gives canonical representations 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 [AL93], and the second to van Oostrom [Oos96].

The '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 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 to belong to the same family, as labels of such redexes are the same. Such redexes cannot be related by the zig-zag if one works with the original system [AL93], but can be related if one works in the refinement of the original system [Oos96]. 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. 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 6 will shed further light on the separability problem.

Definition 3.1 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 $P \approx_S P'$, then Pv is canonical iff so is $P'v$. So canonical forms we speak of are actually objects $\langle P \rangle_S v$, for standard finite reductions P. Our extraction algorithm, defined in Definition 3.4 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. In order for the extraction procedure to be decidable and imply that of the zig-zag relation, we need to establish the decidability of P-neededness for finite reductions P is ASDRSs.

Proposition 3.2 *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. □

Lemma 3.3 *Let $Q : t \stackrel{P}{\twoheadrightarrow} s \stackrel{u}{\rightarrow} e$, where u does not create $v \in e$. Then there is a standard $Q' \approx_L Q$ such that $Q' : t \stackrel{P'}{\twoheadrightarrow} s' \stackrel{u'}{\rightarrow} e$, where $P'u' \trianglelefteq_z Pu$ and u' does not create v.*

Let $\langle P \rangle_S v$ not be canonical. By Definition 3.1, there is $Q : t \stackrel{P'}{\twoheadrightarrow} e \stackrel{u}{\rightarrow} s$ such that $Q \approx_L P$ and v is a u-residual of some $v' \in e$. By Lemma 3.3, Q can be chosen standard. In Q, u does not 'contribute' to v, and in the search for a shortest reduction that creates a redex in the zig-zag class of Qv, contraction of u can be omitted – $P'v' \simeq_z Pv$ and $|P'| < |P|$, since all standard Lévy-equivalent reductions have the same (minimal) length [KG96]. Obviously, reductions creating a redex in some family in a 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 $Pv \stackrel{u}{\rightharpoonup} P'v'$, or just $Pv \rightharpoonup P'v'$. For example, consider the ASDRS corresponding to the TRS $R = \{f(x) \rightarrow g(x),\ a \rightarrow b,\ b \rightarrow c\}$, let $P : f(a) \rightarrow g(a) \rightarrow g(b)$, and let $v = b$ in $g(b)$. Then Pv is not canonical, as $P \approx_S Q : f(a) \rightarrow f(b) \rightarrow g(b)$ and the last step of Q does not create v – the latter is the residual of $v' = b$ in the final term of $P' : f(a) \rightarrow f(b)$. Hence we can perform an extraction step $Pv \rightharpoonup P'v'$. The latter redex is in extraction normal form.

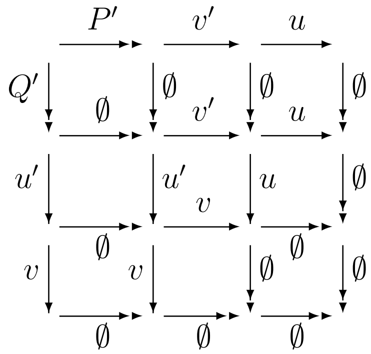
The formal definition of extraction is as follows:

Definition 3.4 (Extraction) Let $Q : t \stackrel{P'}{\twoheadrightarrow} e \stackrel{u}{\rightarrow} s$ 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 $Pv \stackrel{u}{\rightharpoonup} P'v'$, and call the transformation an *extraction* step. {**John:** To exclude: (Note that, since Q is standard, so is P' by DrD.red.need.ess..)} $\rightharpoonup\!\!\!\rightharpoonup$ 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 proof that it is confluent (modulo

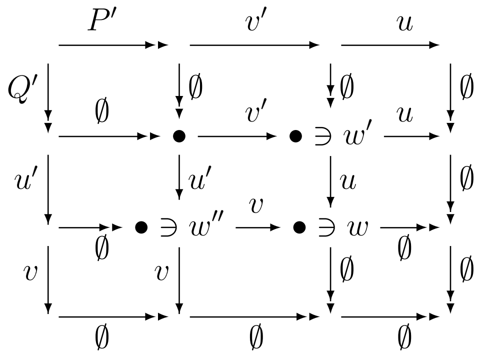
$\approx_S$ on histories), it is enough to prove that it is weakly confluent, that is, $Qw''\stackrel{v}{\leftharpoondown}Nw\stackrel{u}{\rightharpoonup}Pw'$ implies $Qw''\stackrel{u'}{\rightharpoonup}N^*w^*\stackrel{v'}{\leftharpoondown}Pw'$. We need a lemma first {**John:** To exclude: whose proof is delayed to Appendix B,} to enable the reader to follow the main intuition.

Lemma 3.5 *Let $P+u \approx_S Q+v$ and let $u \neq v$. Then there are $P'v'$ and $Q'u'$ such that $P'+v' \approx_S P$, $Q'+u' \approx_S Q$, $P' \approx_S Q'$, $P'v' \simeq_z Qv$ and $Q'u' \simeq_z Pu$, $u = u'/v'$ and $v = v'/u'$.*



Theorem 3.6 *(**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 $\rightharpoonup$ is weakly confluent. Let $Qw''\stackrel{v}{\leftharpoondown}Nw\stackrel{u}{\rightharpoonup}Pw'$ with $u \neq v$ (since if $u = v$ then there is nothing to prove). We will show that $Qw''\stackrel{u'}{\rightharpoonup}N^*w^*\stackrel{v'}{\leftharpoondown}Pw'$ for some N^*w^*, u', and v' such that $u = u'/v'$ and $v = v'/u'$. By Definition 3.4, we have from $Qw''\stackrel{v}{\leftharpoondown}Nw\stackrel{u}{\rightharpoonup}Pw'$ that $Q+v \approx_S N' \approx_S P+u$, where N' is a standard variant of N, and $w''/v = w'/u = w$. By Lemma 3.5, we have the following situation, where $P'+v' \approx_S P$, $Q'+u' \approx_S Q$, $P' \approx_S Q'$, $u = u'/v'$, 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 $Qw''\stackrel{u'}{\rightharpoonup}N^*w^*\stackrel{v'}{\leftharpoondown}Pw'$ by Definition 3.4. □

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

Proof. *By definition of $\simeq_z$, $Pu \simeq_z Qv$ implies existence of $P_0u_0 = Pu, P_1u_1, \ldots P_nu_n = Qv$ 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: $P_1u_1 \rightharpoondown\!\!\!\rightharpoondown P_1'u_1'$. Then there is P_1^* such that $P_1 \approx_S P_1' + P_1^*$. 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'.* {**John:** To exclude: By LrL.era.and.ess.im.ne..(3), }*w does not create the next step in P_1' (if w is not the last step in P_1'), therefore can be permuted with its next step. That w-step is again P_1'-unneeded, 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 $\approx_S$). Since u_1' has a residual along $P_1^* + Q_1$, it is P_0-essential*{**John:** To exclude: by DrD.red.need.ess.}*. Since w is P_0-unneeded, it is P_0-inessential*{**John:** To exclude: by LrL.era.and.ess.im.ne..(1)}*. Hence w does not create u_1'*{**John:** To exclude: by LrL.era.and.ess.im.ne..(3)}*. But this is impossible since $P_1'u_1'$ is canonical and w is the last step of P_1'. So we have proven that P_1' is P_0-needed. This implies that the standardization procedure*{**John:** To exclude: of DrD.ess.rel.red.} *does not effect P_1' when applied to $P_1' + P_1^* + Q_1$, i.e., we can assume a standard $P_0' \approx_S P_0$ such that $P_0' = P_1' + P_0''$ for some P_0'', and $u_0 = u_1'/P_0''$. Hence $P_0u_0 \rightharpoondown\!\!\!\rightharpoondown P_1'u_1'$ by the definition of $\rightharpoondown$, 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. The theorem now follows from Theorem 3.6.* □

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.

4 Affine Zig-zag 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.

Below, $FAM_z(P)$ (resp. $SFAM_z(P)$) denotes the set of zig-zag classes whose member (P-needed) 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 4.1 *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 $P \approx_S Q$, 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 4.2 *If $Pv \simeq_z Qw$, then $v/(Q/P) = w/(P/Q)$. In particular, if $P \approx_S Q$, then $v = w$.*

Proof. *By Theorem 3.7, $Q \approx_L N+Q'$, $P \approx_L N+P'$, $w = u/Q'$ 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 $w/(Q/P) = w/(Q'/P') = u/(P' \sqcup Q') = u/(Q' \sqcup P') = v/(Q'/P') = v/(Q/P)$.* □

Lemma 4.3 *Let $Q^* : t \overset{P}{\twoheadrightarrow} s \overset{u}{\to} e$ and u create $v \in e$. Then, for any canonical form $Q'v'$ of Q^*v, Q' contracts a redex zig-zag related to Pu.*

Proof. *We have by Lemma 3.3 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 3.3 there is an extraction step $Qv \overset{w_1}{\rightharpoonup} Q_1 v_1$ (i.e., $Q \approx_S Q_1 + w_1$ and $v = v_1/w_1$). Since $Q_1 + w_1 \approx_S Q = P' + u'$, we have by Lemma 3.5 that $Q_1 \approx_S P_1 + u_1$ such that $P_1 u_1 \simeq_z P'u' \simeq_z Pu$. So we have $(P' + u')v \overset{w_1}{\rightharpoonup} (P_1 + u_1)v_1$ 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 $(P_1 + u_1)v_1 \overset{w_2}{\rightharpoonup} (P_2 + u_2)v_2$ 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 3.6, for any canonical form $Q'v'$ of Qv (and hence of Q^*v), $Q' \approx_S P_m + u_m$ and $v_m = v'$, it follows by Lemma 4.1 that Q' contracts a redex in the family of Pu.* □

Lemma 4.4 *Let $Pv \overset{w}{\rightharpoonup} P'v'$. Then $FAM_z(P') \subseteq FAM_z(P)$.*

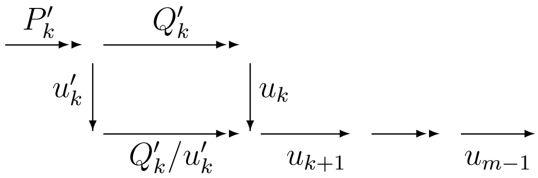
Proof. *By Definition 3.4, $Pv \overset{w}{\rightharpoonup} P'v'$ implies that $P' + w \in STV(P)$, and by Lemma 4.1, $FAM_z(P') \subseteq FAM_z(P' + w) \subseteq FAM_z(P)$.* □

Lemma 4.5 *Let $Q : e \overset{P}{\twoheadrightarrow} t \overset{u}{\to} s$ and let u create $v \in s$. Then $Fam_z(Pu) \hookrightarrow_z Fam_z(Qv)$.*

Proof. *By Lemma 4.3, if $Q'v'$ is a canonical form of Qv, then $Fam_z(Pu) \in FAM_z(Q')$. Now it follows from Lemmas 4.1 and 4.4 and Theorem 3.7 that for any $Q^*v^* \simeq_z Qv$, $Fam_z(Pu) \in FAM_z(Q^*)$, i.e., $Fam_z(Pu) \hookrightarrow_z Fam_z(Qv)$.* □

Lemma 4.6 *Let $P : t_0 \overset{P_i}{\twoheadrightarrow} t_i \overset{u_i}{\to} t_{i+1} \twoheadrightarrow t_n$. Then $k < m$ implies $Fam_z(P_k u_k) \neq Fam_z(P_m u_m)$.*

Proof. *By induction on the number of zig-zag classes $\hookrightarrow_z$-contributing to $Fam_z(P_k u_k)$. Suppose on the contrary that $Fam_z(P_k u_k) = Fam_z(P_m u_m)$. Let $P'_k u'_k$ be a canonical form of $P_k u_k$, which exists by Theorem 3.6, 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 $P_{k+1} = P_k + u_k \approx_L P'_k + Q'_k + u_k \approx_L P'_k + u'_k + Q'_k/u'_k$. Since P'_k contracts redexes in all contributor zig-zag classes of $Fam_z(P'_k u'_k) = Fam_z(P_k u_k) = Fam_z(P_m u_m)$, and since by the induction assumption no redexes in these classes 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 Lemma 4.5. 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 $Fam_z(P'_k u'_k) = Fam_z(P_m u_m) = Fam_z(P'_k u'_m)$ and $u'_k \neq u'_m$, which is not possible by Lemma 4.2 – contradiction.*



□

Theorem 4.7 *Let $\mathcal{R}$ be a non-duplicating stable DRS. Then $\mathcal{F}_\mathcal{R} = (R, \simeq_z, \hookrightarrow_z)$ is a zig-zag DFS.*

Proof. *We need to show that $\mathcal{F}_\mathcal{R}$ 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 3.7 that $\emptyset_t u \not\simeq_z \emptyset_t v$, i.e., [initial] holds. [creation] is immediate from Lemma 4.5, and [FFD] from Lemma 4.6.*□

5 Affine Separable Families

Based on the results of previous sections, we now 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*. Recall that a complete family-reduction is a multi-step reduction contracting, in each multi-step, all members of a single family in parallel (we will often omit 'complete').

Lemma 5.1 *Let $\mathcal{F} = \mathcal{R}, \simeq, \hookrightarrow)$ be an affine DFS. Them 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 family-reduction, and vice versa.

Proof. (1) $\Rightarrow$ (2) *Similar to the proof of Lemma 4.6, except replacing $Fam_z(\ )$ by $Fam(\ )$, using [creation] instead of Lemma 4.5, and using separability instead of Lemma 4.2 (see [KG97]).*

(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 family-reduction,*

implying that $P + v + w' \| w''$, where $w' \| w''$ is the multi-step contracting w' and w'' in parallel, is a family-reduction which is not a reduction, contradicting (4). □

Lemma 5.2 *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 3.1, 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* {**John:** To exclude: (by LrL.era.and.ess.im.ne.)} *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.* □

Theorem 5.3 *An affine DFS $\mathcal{F}$ is separable iff it is a zig-zag DFS.*

Proof. *($\Rightarrow$) Let $\mathcal{F} = (\mathcal{R}, \simeq_s, \hookrightarrow_s)$ be separable, and let $Pu \simeq_s Qv$. (We want to prove that $Pu \simeq_z Qv$.) By the Extraction Theorem, there are canonical forms $P'u' \simeq_z Pu$ and $Q'v' \simeq_z Qv$, and since $\simeq_z \subseteq \simeq_s$ (by the definition of DFSs), $P'u' \simeq_s Q'v'$. Now if $P' \approx_S Q'$, since both $P'u'$ and $Q'v'$ are canonical, the last step of P' creates both u' and v', and $u' = v'$ by separability. Hence, $P'u' \simeq_z Q'v'$, implying $Pu \simeq_z Qv$, in this case. Otherwise, we must have $P' \not\approx_L Q'$ (as $P', Q' \in STA$), and if say $P' \lhd Q'$, $Q' \approx_L P' + P''$ for some $P'' \neq \emptyset$. Again, since $P'u'$ and $Q'v'$ are canonical, the last step of P' creates u', and that of P'', call it w, creates v' (since the last step of P'' coincides with that of $ST(P' + P'')$). By [creation], $Fam(w) \hookrightarrow_s Fam(v') = Fam(u')$, hence by [contribution] P' must also contract a redex in $Fam(w)$, contradicting separability by Lemma 5.1. Hence $P' \not\approx_L Q'$ cannot hold, and we are done. ($\Leftarrow$) Let $\mathcal{F}$ be a ZDFS. By Lemma 4.6, any family is contracted at most once in a reduction in $\mathcal{F}$, implying by Lemma 5.1 that $\mathcal{F}$ is an SDFS.* □

We conclude this section by a useful characterization of histories of canonical elements of zig-zag (hence extraction and separable) families, in ASDRSs.

Theorem 5.4 *Let Pv be a canonical element of a family ϕ, in an AZDFS $\mathcal{F}_{\mathcal{R}} = (R, \simeq_z, \hookrightarrow_z)$. Then P contracts exactly one redex in every contributor family of ϕ.*

Proof. *By Lemma 5.2, P contracts only redexes in contributor families of ϕ. That every such a family is contracted in P follows immediately from [contribution]. The uniqueness of contracted families follows from Lemma 5.1 and Theorem 5.3.* □

6 Implementation DFSs

We now define the *implementation* $\mathcal{F}_I$ of a DFS $\mathcal{F}$, whose reductions 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 $\mathcal{S}$-normalizing reductions. We will assume that the reduction graph of $\mathcal{F}$ is the reduction graph of an *initial* term, denoted by $\emptyset$, and that families are considered relative to $\emptyset$, i.e, all histories start with $\emptyset$.

Definition 6.1 Let $\mathcal{F} = (\mathcal{R}, \simeq, \hookrightarrow)$ be a DFS. 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 of $\mathcal{R}_I = (A, /)$ are family-reductions starting from $\emptyset$, each edge (i.e., a reduction step) being a multi-step contracting a family of redexes.

• the residual relation / is defined as follows: let U and V be complete sets of redexes in two families, in a term s, and let $U : s \to o$ be the multi-step contracting U. Then V/U is the multi-step $o \to e$ 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 family-reductions P and Q in $\mathcal{F}$; then $P_I U \simeq_I Q_I V$ iff for any $u \in U, v \in V$, $Pu \simeq Qv$; and $Fam(P_I U) \hookrightarrow_I Fam(Q_I V)$ iff $Fam(Pu) \hookrightarrow Fam(Qv)$.

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

Lemma 6.2 *Let $P : \emptyset \twoheadrightarrow s$ be a 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 family-reduction), and $P + V$ contracts the family ψ twice*{**John:** To exclude: , contradicting LrL.fam.compl.}. □

Thus the residual relation is well defined. Obviously, $V/V = \emptyset$, 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 Acyclicity Lemma and 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 family-reductions can be viewed as reduction 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 {**John:** To exclude: from LrL.fam.compl.} 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 5.3:

Theorem 6.3 *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 6.4 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 6.5 *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 (any) reductions in $\mathcal{R}$ corresponding to P and Q, and v' and u' are any $\mathcal{R}$-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 $\simeq^*$, family-reductions in $\mathcal{F}^*$ are exactly family-reductions in $\mathcal{F}$, and $\mathcal{F}^*_I = \mathcal{F}_I$ follows since both are AZDFSs by Theorem 6.3. 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 5.3. ($\Leftarrow$) Immediate from Definition 6.4.* □

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 lemma{**John:** To exclude: , whose proof can be found in Appendix B,} relates neededness in a DFSs with neededness in its implementation. {**John:** To exclude: Together with TrT.unif.opt., }It implies that the implementation DRSs $\mathcal{F}_I$ indeed correctly implements family-reductions in DFSs $\mathcal{F}$.

Lemma 6.6 *Let $\mathcal{S}$ be a stable set of terms* {**John:** To exclude: (see DrD.rel.need.)} *in a DFS $\mathcal{F}$ not containing the initial term $t_0 = \emptyset$, let $P : t_0 \overset{U_0}{\twoheadrightarrow} t_1 \overset{U_1}{\twoheadrightarrow} \ldots \twoheadrightarrow t_n$ be an $\mathcal{S}$-normalizing family-reduction in $\mathcal{F}$, and let $P_I : t_0 \overset{u_0}{\rightarrow} t_1 \overset{u_1}{\rightarrow} \ldots \rightarrow t_n$ be its corresponding reduction in $\mathcal{F}_I$. Then P is $\mathcal{S}$-needed iff so is P_I.*

Theorem 6.7 ***(Optimal Implementation)*** *Let $\mathcal{S}$ be a stable set of terms* *in a DFS $\mathcal{F}$. Then optimal (i.e., $\mathcal{S}$-needed) reductions in $\mathcal{F}_I$, w.r.t. $\mathcal{S}$, implement optimal family-reductions in $\mathcal{F}$, w.r.t. $\mathcal{S}$.*

Proof. *$\mathcal{S}$-needed reductions in $\mathcal{F}_I$, which actually are $\mathcal{S}$-needed family-reductions by Lemma 5.1.(4) and Theorem 6.3, and hence are optimal, implement optimal family-reductions in $\mathcal{F}$.* □

7 Conclusions and Future Work

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 the original DFSs. Further, 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 simultaneously two different members of the same family. Finally, we have shown that sharing is compositional. In particular, any family relation can be decomposed into the 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 [AL93]).

The optimality theory is not the only concern in this work. Non-duplicating systems are of great importance for the study of semantics of computation. Recall that say *distributive domains* (also called *dI-domains* or *stable domains*) correspond to *linear* systems, where there is no duplication nor erasure of redexes [Win86/89].

Indeed, the results on ASDRSs established in this work are the basis of our investigation of the semantics of orthogonal systems. To the best of our knowledge, DFSs are the only abstract systems that allow one to project duplicating and/or erasing computation onto non-duplicating computation, and indeed the results in this paper prove the correctness of such projection. It is shown in [KG02] that ASDRSs can be seen as the refinement of distributive domains (in the conflict-case): they are distributive domains enriched by an axiomatized *erasure* relation. In these systems, distributivity can be restored by considering needed reductions only.

These projection results have also been used in the definition of *independence* of computations, and construction of Euclidian Geometry from the reduction spaces in orthogonal rewrite systems [KG97a]. The projection results allow to prove results decomposition, normalization and optimality results for duplicating erasing systems by performing proofs for ASDRSs, where proofs are of course much simpler.

We also refer to [KG03] for recent results on computational semantics of conflict free reduction systems, where the framework of SDRSs and DFSs is used as the basis. There, new partial orders, more refined than those of distributive domains are introduced and studied, and relevance of results obtained in this paper for building more refined lambda models from these orderings is an important open question.

References

[AL93] Asperti A., Laneve C. Interaction Systems I: the theory of optimal reductions. MSCS, 11:1-48, Cambridge University Press, 1993.

[AG98] Asperti A., Guerrini, S. The Optimal Implementation of Functional Programming Languages (Cambridge Tracts in Theoretical Computer Science 45, Cambridge University Press, 1998).

[Ber79] Berry G. Modéles complétement adéquats et stables des λ-calculs typés. Thèse de l'Université de Paris VII, 1979.

[GK96] Glauert J.R.W., Khasidashvili Z. Relative normalization in deterministic residual structures. CAAP'96, Springer LNCS, vol. 1059, H. Kirchner, ed. 1996, p. 180-195.

[GK02] Glauert J.R.W., Khasidashvili Z. An Abstract Böhm normalization. WRS'02, vol 70 (6), Elsevier Science B.V., 2002.

[GLM92] Gonthier G., Lévy J.-J., Melliès P.-A. An abstract Standardisation theorem. LICS'92, p. 72-81.

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

[HP91] Hoffmann B., Plump D. Implementing term rewriting by Jungle evaluation. Theoretical Informatics and Applications 25:445-472, 1991.

[KKSV93] Kennaway J. R., Klop J. W., Sleep M. R, de Vries F.-J. Event structures and orthogonal term graph rewriting. In: M. R. Sleep, M. J. Plasmeijer, and M. C. J. D. van Eekelen, eds. Term Graph Rewriting: Theory and Practice. John Wiley, 1993.

[KG96] Khasidashvili Z., Glauert J. R. W. Discrete normalization in Deterministic Residual Structures. ALP'96, Springer LNCS, 1139, M. Hanus, M. Rodríguez-Artalejo, eds. 1996, p.135-149.

[KG97] Khasidashvili Z., Glauert J. R. W. Zig-zag, extraction and separable families in stable non-duplicating Deterministic Residual Structures. Technical Report IR-420, Free University, Amsterdam, February 1997.

[KG97a] Khasidashvili Z., Glauert J. R. W. The Geometry of orthogonal reduction spaces. ICALP'97, Springer LNCS, vol. 1256, P. Degano, R. Gorrieri, and A. Marchetti-Spaccamela, eds. 1997, p. 649-659. Full version to appear in Theoretical Computer Science.

[KG02] Khasidashvili Z., Glauert J. R. W. Relating conflict-free transition and event models via redex families. Theoretical Computer Science, vol. 286, 2002, p. 65-95.

[KG03] Khasidashvili Z., Glauert J. R. W. Stable computational semantics of conflict-free rewrite systems (partial orders with erasure). RTA'03 (to appear). Available at http://www.sys.uea.ac.uk/ jrwg/papers/rta03subJG.pdf.

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

[Lév78] Lévy J.-J. Réductions correctes et optimales dans le lambda-calcul, Thèse de l'Université de Paris VII, 1978.

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

[Mar91] Maranget L. Optimal derivations in weak λ-calculi and in orthogonal Term Rewriting Systems. POPL'91, p. 255-269.

[Mel0+] Melliès P.-A. Axiomatic Rewriting Theory I. A diagrammatic standardization theorem Submitted. Available at http://www.pps.jussieu.fr/˜mellies/.

[Oos96] Van Oostrom V. Higher order families. RTA'96, Springer LNCS, vol. 1103, Ganzinger, H., ed., 1996, p. 392–407.

[Plo75] Plotkin G. Call-by-name, call-by-value and the λ-calculus. TCS(1):125-159, 1975.

[Sta89] Stark E. W. Concurrent transition systems. TCS 64(3):221-270, 1989.

[Ter03] Terese. Term Rewriting Systems. Cambridge Tracts in Theoretical Computer Science, Volume 55, Cambridge University Press, 2003.

[Wad71] Wadsworth C.P. Semantics and pragmatics of the Lambda-calculus. Ph.D. thesis, University of Oxford, 1971.

[Win86/89] Winskel G. An introduction to Event Structures. In proc. of School/Workshop on 'Linear time, branching time and partial order in logics and models of concurrency'. Springer LNCS, vol. 354, 1989, p. 364-397. Extended version appears as Cambridge University Computer Laboratory Report n. 95, 1986.

8 Appendix A: Deterministic Residual and Family Structures

In this section we recall *Deterministic Residual Structures* (DRSs) and *Deterministic Family Structures* (DFSs), and give a number of examples. We will also recall some basic theory of standardization in DRSs from [KG96].

The DRS concept is based on [Lév80,HL91]. DRSs model orthogonal term as well as graph rewrite systems, both first and higher order, and including the λ-calculus and its sharing evaluation models, with the standard Church notion of residual. Closely related models are the CTSs of Stark [Sta89], the ARSs of Gonthier et al. [GLM92], and Axiomatic Rewriting Theory of Melliès [Mel0+]. See [Ter03] for more information on abstract rewriting (with or without a residual relation). DRSs are a minimal axiomatization of the residual concept enabling one to develop a theory of normalization [GK96] and standardization [KG96] for conflict-free reduction systems in an abstract manner, and to study their semantics [KG02,KG97a].

Definition 8.1 A 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. 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 u/P of residuals of u 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] ([GLM92]) 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] ([Sta89]) Let $u, v \in t$, let $u \neq v$, and let $u/v = \emptyset$. Then $v/u \neq \emptyset$.

We call a DRS $\mathcal{R}$ *stable* (SDRS) if:

• [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 .

We call an SDRS *non-duplicating* or *affine*, ASDRS, if the residual relation is non-duplicating. Note that, since the only observables of DRSs are redexes, duplicating syntactic rewrite systems may still form affine SDRSs. For example, the DRS corresponding to innermost reductions in an orthogonal TRS is an affine SDRS, although innermost redexes may duplicate their arguments.

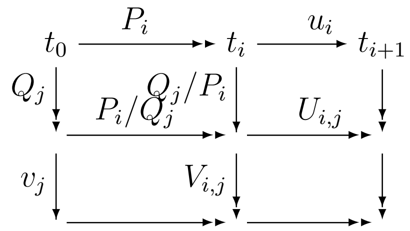
Similarly to [HL91,Lév78,Lév80,Sta89], in a DRS $\mathcal{R}$ the residual relation on redexes is extended to all co-initial reductions as follows: $(P_1 + P_2)/Q = P_1/Q + P_2/(Q/P_1)$ and $P/(Q_1 + Q_2) = (P/Q_1)/Q_2$, and that *Lévy-equivalence* or *permutation-equivalence* is defined as the smallest relation on co-initial reductions satisfying: $U + V/U \approx_L V + U/V$ and $Q \approx_L Q' \implies P + Q + N \approx_L$

$P + Q' + N$, where U and V are complete developments of redex sets in the same term. Further, one defines $P \trianglelefteq Q$ iff $P/Q = \emptyset$, and can show that $P \approx_L Q$ iff $P \trianglelefteq Q$ and $Q \trianglelefteq P$; and $P \trianglelefteq Q$ iff $Q \approx_L P + N$ for some N. Below, $P \sqcup Q$ will denote $P + Q/P$. Intuitively, $P \approx_L Q$ means that P can be obtained from Q by a number of permutations of adjacent steps, therefore 'Q and P do the same work'; and $P \trianglelefteq Q$ means that P does less work than Q, the difference being Q/P, so $P + Q/P \approx_L Q$. The following *Strong Church-Rosser* property can be proved: for any co-initial finite reductions P, Q, $P \sqcup Q \approx_L Q \sqcup P$.

The stability axiom, and more generally Lemma 8.3 below, states that a redex cannot arise from two 'unrelated' sources. The notion of 'unrelated' is formalized by the notion of *externality*, which expresses the absence of shared (residuals of) redexes.

Definition 8.2 ([GK96]) • Let $u \in U \subseteq t$ and $P : t \twoheadrightarrow o$. We call P *external* to {**John:** redexes} U (resp. u) if P does not contract residuals of redexes in U (resp. residuals of u).

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



Lemma 8.3 (Stability) *([GK96]) 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.*

Lemma 8.4 (Weak Acyclicity) *([KG96]) Let P, N be external co-initial finite reductions in a DRS. Then $N \not\approx_L P$.*

8.2 Standardization

In this section, we recall a fragment of results from [KG96], concerning standardization of reductions, relevant to this paper; so we restrict ourselves to affine stable DRSs, ASDRSs, only.

Definition 8.5 • 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 call u P*-essential* if there is no $Q \approx_L P$ that discards u, and P*-inessential* otherwise.

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

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

Note that P-neededness, P-erasure, and P-essentiality 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 8.6 *Let $P : s \twoheadrightarrow t \stackrel{u}{\to} e \twoheadrightarrow o$ in an ASDRS.*

(1) $w \in t$ is P-needed iff it is P-erased and P-essential.

(2) If $P : t \twoheadrightarrow s' \stackrel{w}{\to} o$, then $w \in s'$ is P-needed.

(3) If u creates $v \in e$ and u is P-unneeded (resp. P-inessential), then so is v.

(4) If $u \neq v \in t$, then v is P-needed (P-essential) iff v has a P-needed (P-essential) residual in e.

Self-needed reductions play the role of *standard* reductions in SDRSs, since we do not have any nesting relation imposed on redexes, unlike ARSs of [GLM92], and there is no concept of 'left' or 'right' occurrences in DRSs. In the extraction process which we study below (for ASDRSs), self-needed reductions play the same role as outside-in left-to-right standard reductions in the extraction processes of [Lév80,AL93,Oos96].

Definition 8.7 We call a reduction in a DRS *standard* if it is self-needed. We write $P \approx_S Q$ if $P \approx_L Q$ and both P and Q are standard. For any standard P, we define $\langle P \rangle_S = \{Q \mid Q \approx_S P\}$.

The following algorithm is a standardization procedure for reductions in ASDRSs.

Definition 8.8 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 8.6). Then $ST(P) = v + ST(P/v)$.

Termination of the standardization follows immediately from the fact that $|P/v| \leq |P| - 1$. The following fragment (for ASDRSs) of the standardization theorem from [GK96] implies the correctness of our standardization procedure.

Theorem 8.9 (Standardization) *For any finite reduction P in a stable non-duplicating DRS, $ST(P)$ is a finite standard reduction Lévy-equivalent to P.*

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 8.6 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 8.6.(2). If it is created by u_{n-2}, then the latter is P-needed too, by Lemma 8.6.(3). Otherwise, the ancestor redex of u_{n-1} in t_{n-2} is P-needed by Lemma 8.6.(4). 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; this can be shown exactly as the correctness of the standardization algorithm (and is also a corollary of the *Discrete Normalization Theorem* of [KG96]).

In Section 3, we will show that, moreover, all standard variants of a finite reduction P in an ASDRS can be found effectively (there are clearly only a finite number reductions in $STV(P)$ as they all have the same length, which coincides with the number of P-needed steps in P).

8.3 Deterministic Family Structures

We now recall *Deterministic Family structures* (DFSs) which are DRSs where in addition a notion of *redex-family* is axiomatized so that the essence of sharing in the sense of Lévy [Lév78,Lév80] is captured, and all the known family notions (mentioned in the introduction) satisfy these axioms [GK96].

Definition 8.10 A DFS is a triple $\mathcal{F} = (\mathcal{R}, \simeq, \hookrightarrow)$, where $\mathcal{R}$ is a DRS; $\simeq$ 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, written $Pu \trianglelefteq_z Qv$, if $P \trianglelefteq Q$, 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 [Lév80]. 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) The relations $\simeq$ 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] If $e \overset{P}{\twoheadrightarrow} t \xrightarrow{u} s$ and u creates $v \in s$, then $Fam(Pu) \hookrightarrow Fam((P+u)v)$.

• [FFD] (*Finite Family Developments*) Any reduction that contracts redexes of a finite number of families is terminating. [4]

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 $\mathcal{F} = (\mathcal{R}, \simeq)$.

We will need the following results from [GK96].

[4] This axiom was called [termination] in [GK96].

Lemma 8.11 *(***Unique Families***) Every family is contracted at most once in a (complete) family-reduction, in a DFS.*

Definition 8.12 ([GK96]) (1) Let $\mathcal{S}$ be a set of terms in a DRS. 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 it $\mathcal{S}$*-unneeded* otherwise. We call a multi-step $U \subseteq t$ $\mathcal{S}$-needed if it contracts at least one $\mathcal{S}$-needed redex.

(2) We call a set $\mathcal{S}$ of terms *stable* if: (a) $\mathcal{S}$ is *closed under reduction*; and (b) $\mathcal{S}$ is *closed under unneeded expansion*: for any $e \overset{u}{\to} o$ such that $e \notin \mathcal{S}$ and $o \in \mathcal{S}$, u is $\mathcal{S}$-needed.

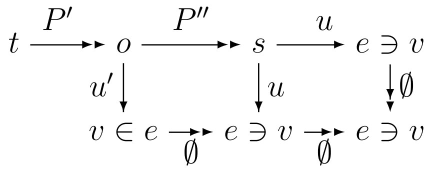
Theorem 8.13 (Relative Optimality [GK96]) *Let $\mathcal{S}$ be a stable set of terms in a DFS $\mathcal{F}$, and let t be an $\mathcal{S}$-normalizable term in $\mathcal{F}$. Then any $\mathcal{S}$-needed (complete) family-reduction Q starting from t is eventually $\mathcal{S}$-normalizing and is $\mathcal{S}$-optimal in the sense that it has a minimal number of family-reduction steps.*

Appendix B

Proof of Proposition 3.2
Let $P : t_0 \overset{u_0}{\to} t_1 \overset{u_1}{\to} \ldots \to t_n$. 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 8.5 and Lemma 8.6.(2). Let $n > 1$, and let $P_1 : t_1 \overset{u_1}{\to} t_2 \overset{u_2}{\to} \ldots \to t_n$. 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 8.5 (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 8.6.(4). 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 8.5; 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 8.5), then u_0 is P-needed by Lemma 8.6.(3); if not, then we arrive to a previously considered case, and the decidability of P-neededness follows. The rest follows from the correctness of the construction of a standard variant of P discussed in Subsection 8.2.

Proof of Lemma 3.3 We show that $ST(Q)$ can be taken for Q'. By Definition 8.8, $ST(Q)$ 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 8.6.(2), 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 8.6.(3), P'' cannot create u, i.e., there is

$u' \in o$ such that $u'/P'' = u$, and u' is Q-needed by Lemma 8.6.(4). Since P''/u' is Q-unneeded by Lemma 8.6.(4), and since the last step of $P' + u' + P''/u'$ is Q-needed by Lemma 8.6.(2), $P''/u' = \emptyset$. Since u' is Q-needed and P'' is Q-unneeded, P'' is external to u' by Lemma 8.6.(4). Hence, by the Stability Lemma, u' does not create v, and the lemma follows since $ST(Q) = P' + u'$ is standard by Theorem 8.9, and $P'u' \trianglelefteq_z Pu$ since $u = u'/P''$.



Proof of Lemma 6.6 ($\Rightarrow$) Assume that P be $\mathcal{S}$-needed, and suppose on the contrary that P_I is not $\mathcal{S}$-needed, i.e., u_i is $\mathcal{S}$-unneeded for some i. Then there is an $\mathcal{S}$-normalizing reduction $N : t_i \twoheadrightarrow s$ that is external to u_i. It remains to show that any corresponding 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 6.2 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 $\mathcal{S}$-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 family-reduction (which exists by Theorem 8.13). Note that the residual of U_i in any term along Q forms a complete family by Lemma 6.2, and all residuals of redexes in U_i remain $\mathcal{S}$-unneeded by Corollary 3.1 of [GK96] (which states exactly that the residuals of $\mathcal{S}$-unneeded redexes remain $\mathcal{S}$-unneeded). 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 $\mathcal{S}$-neededness of u_i.

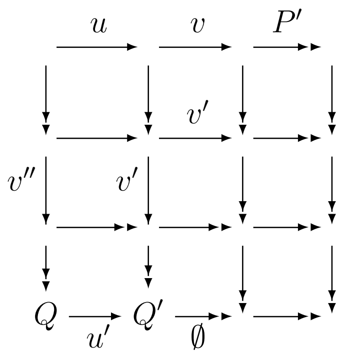
Appendix C: Proof of Lemma 3.5

We use the following lemma from [GK96], and three simple new lemmas in this proof.

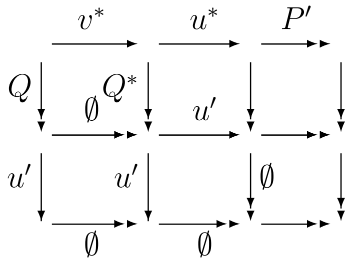
Lemma 8.14 *([GK96]) Let $P : t \twoheadrightarrow s$ be external to a set $U \subseteq t$ of redexes, in a DRS, and let $Q : t \twoheadrightarrow o$. Then P/Q is external to the set U/Q.*

Lemma 8.15 *Let $u + P \approx_S Q + u'$, 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 8.5, 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:*



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 8.14 that $Q^* = Q/v^*$ is external to u^*. Since u is $u + P$-needed, so is u^* by Lemma 8.6.(4). Hence u^* is $Q^* + u'$-needed. Since Q^* is external to u^* and $Q^* + u'$ contracts a residual of u^*, we have $u' = u^*/Q^*$.



□

Lemma 8.16 *Let $P + u \approx_S Q + v$ and let $u \neq v$. Then $P \not\approx_L Q$.*

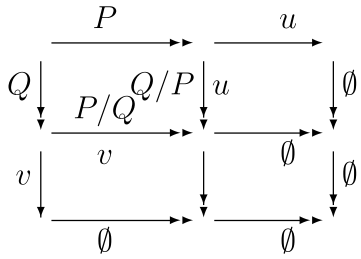
Proof. *Suppose on the contrary that $P \approx_L Q$. Then $P+u \approx_S Q+v$ iff $u \approx_L v$. But, by [weak acyclicity], this is only possible when $u = v$ – a contradiction.* □

Lemma 8.17 *Let $P \approx_S Q$. Then any non-empty step in Klop's diagram of P and Q in P-needed.*

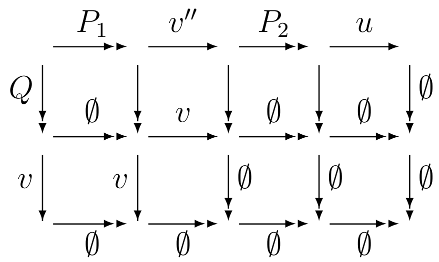
Proof. *Since every step in the diagram is a residual of a redex contracted in P or Q, the lemma follows immediately from Lemma 8.6.(4).* □

Proof of Lemma 3.5 Since $P + u \approx_S Q + v$, we have $v \approx_L (P + u)/Q$. By

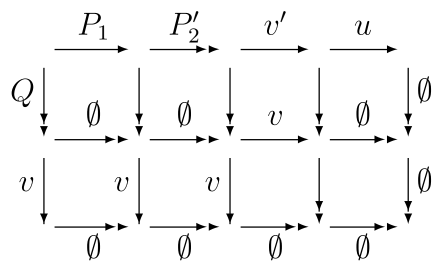
Lemma 8.4, $(P+u)/Q$ contracts a residual of v. We show that $P/Q \neq \emptyset$.



Suppose on the contrary that $P/Q = \emptyset$. Then, by [weak acyclicity], $v = u/(Q/P)$. Further, by Lemma 8.16, $P \not\approx_L Q$, hence $P/Q = \emptyset$ implies $Q/P \neq \emptyset$. 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 8.17. 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 $v'' + P_2 + u \approx_S Q/P_1 + v$ and $v'' + P_2 \not\approx_L Q/P_1$:

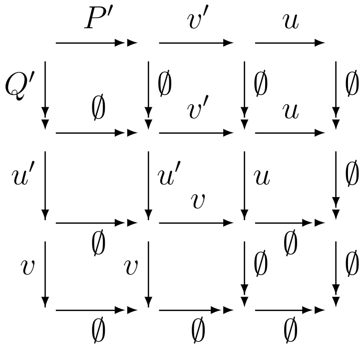


Now, by repeated application of Lemma 8.15, $v'' + P_2 + u$ can be transformed into a reduction $P_2' + v' + u$ such that $v'' + P_2 \approx_S P_2' + v'$, $v' = v''/P_2'$, and $v = v'/(Q/(P_1 + P_2'))$.



Hence, if we take $P' = P_1 + P_2'$, we have that $P' + v' \approx_S P$ and $P'v' \trianglelefteq_z Qv$. Existence of $Q'u'$ such that $Q' + u' \approx_S Q$ 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 8.15 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

$Q'u' \trianglelefteq_z Pu$).



Hence we have by Lemma 8.4 and Lemma 8.17 that $P'/Q' = \emptyset$. The converse is proved similarly, so $P' \approx_L Q'$. It follows that $u = u'/v'$ and $v = v'/u'$.