



TCS4127

pp: 1-31 (col.fig.: nil)

PROD. TYPE: COM

ED: JSS

PAGN: LAKSHMI - SCAN: nil



ELSEVIER

Theoretical Computer Science 000 (2002) 000–000

Theoretical
Computer Science

www.elsevier.com/locate/tcs

1 Relating conflict-free stable transition and event models 3 via redex families[☆]

Zurab Khasidashvili^a, John Glauert^{b,*}^aDepartment of Mathematics and Computer Science, Bar-Ilan University, Ramat-Gan 52900, Israel^bSchool of Information Systems, University of East Anglia, Norwich NR4 7TJ, UK

7 Abstract

9 We describe an event-style (or poset) semantics for conflict-free rewrite systems, including
10 term and graph rewriting (possibly with bound variables), the λ -calculus, and other stable transi-
11 tion systems with a residual relation. Our interpretation is based on considering redex-families
12 as events. It treats permutation-equivalent reductions as representing the same concurrent com-
13 putation. Due to erasure of redexes, event structures are inadequate for such an interpretation.
14 We therefore extend the prime event structure model in two different but equivalent ways: by
15 axiomatizing permutation-equivalence on finite configurations, and by axiomatizing the erasure
16 of events, for the conflict-free case, and show that these extended models are equivalent to sta-
17 ble transition models with axiomatized residual and family relations. We then construct finitary
18 prime algebraic domains from the set of configurations in these extended models by defining
19 orderings relative to stable sets of ‘results’. All useful sets of results for which the normalization
(by neededness) theorem can be proved are stable. © 2002 Published by Elsevier Science B.V.

Keywords: Abstract reduction systems; Redex families; Event structures; Distributive domains

21 1. Introduction

22 The primary goal of this paper is to provide a fully adequate event-style concur-
23 rent semantics for orthogonal rewrite systems. Event structures, developed by Winskel,
Nielsen and Plotkin [32, 36–38], are a commonly accepted model of concurrency.
24 Conflict-free prime event structures (PESs) are sufficient for our purposes. A (conflict-
25 free) PES is simply a set of events partially ordered by a *causal dependency* relation

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

* Corresponding author.

E-mail addresses: khasidz@cs.biu.ac.il (Z. Khasidashvili), jrwg@sys.uea.ac.uk (J. Glauert).

1 (a *poset*) where an event can only dominate a finite number of events (the so-called *ax-*
2 *iom of finite causes*). The idea is to view a process as performing events (some *atomic*
3 tasks), and an event can occur, only once, after all events on which it depends. Causally
4 independent events are *concurrent* events and can be evaluated in parallel. Stages of
5 a computation are thus represented by *configurations*—left-closed event sets w.r.t. the
6 causal order, and the same configuration represents all sequential computations execut-
7 ing events in that configuration, ignoring the order in which the concurrent events occur.

8 In the theory of orthogonal rewrite systems, such as the λ -calculus, there is the well
9 established concept of ‘redex-family’, due to Lévy [26, 27], which formalizes the idea of
10 the ‘same atomic task’. Therefore, it is natural to base our interpretation on considering
11 redex-families as events, and to interpret complete family-reductions (contracting entire
12 families as multi-steps) as configurations of the corresponding PES. This is also justified
13 by the fact that reductions in optimal graph-implementations of orthogonal systems
14 correspond (up to book-keeping steps) to complete family-reductions in the original
15 system [1, 15, 23, 28].¹

16 Recent advances in the abstract study of the syntax and operational semantics of
17 rewrite and transition systems [11, 12, 17, 18, 20, 29, 35] allow us to address the prob-
18 lem in an entirely abstract setting. Stable deterministic residual structures (SDRSs) and
19 Deterministic family structures (DFSs) model computation in orthogonal rewrite sys-
20 tems. DRSs are abstract rewrite systems with an axiomatized residual relation; DFSs
21 are DRSs where in addition the concept of redex-family is axiomatized. Stable DRSs
22 allow [11, 17] proofs of analogs of the normalization and standardization theorems
23 [3]. In DFSs one can further prove the Optimality Theorem and the Unique Families
24 Lemma [11]. The latter states that any family can be contracted at most once in a
25 complete family-reduction. It corresponds to the fact that any event may occur at most
26 once in the course of a computation, and is needed for the event structure interpretation
27 of SDRSs based on interpreting families as events.

28 Thus, we can start our interpretation by assuming that an orthogonal rewrite system
29 is given in the form of a DFS. Furthermore, it has been shown in [20] that any DFS
30 can be given an interpretation by a non-duplicating or affine DFS with zig-zag as the
31 family relation (AZDFS), called the *implementation* of the original DFS: the reduction
32 steps in the implementation DFS correspond to complete family-reduction multi-steps
33 of the original DFS. Therefore, it is enough to concentrate on defining PES semantics
34 for AZDFSs.

35 To view a process as the domain of all its configurations, an *adequacy* requirement
36 is desirable. Namely, that *runs* of a finite configuration (i.e., all its possible sequen-
37 tializations) should be in one-to-one correspondence with all finite reductions in the
38 equivalence class which the configuration is supposed to represent. Because of the
39 huge importance of *permutation* or *Lévy equivalence* to the whole theory of (not only
40 orthogonal) rewrite systems [5, 14, 27], and to concurrency theory as well [35], we
41 want to be able to treat permutation-equivalent finite reductions as representing the

¹ The book-keeping steps can be expensive in the case of λ -calculus [2].

1 same concurrent computation, as indeed permutation-equivalent finite reductions result
 2 one from another by permuting concurrent consecutive steps. However, this is clearly
 3 impossible in the present PES framework [37], as a finite configuration in a PES has
 4 only a finite number of runs while, a permutation-equivalence class of a finite reduction
 5 may have an infinite number of elements.

6 For example, consider the λ -term $t = Kx\Omega$, where $K = \lambda xy.x$ and $\Omega = (\lambda x.x)\lambda x.x$.
 7 Intuitively, all reductions of the form $t \xrightarrow{\Omega} t \xrightarrow{\Omega} \dots \xrightarrow{\Omega} t \xrightarrow{K} x$, or $\Omega^m K$ in short, consisting
 8 of a number of Ω -steps followed by a K -step, are ‘equal’, as the Ω -steps are performed
 9 in an erased part of t (or in ‘garbage’). And indeed, permutation equivalence equates all
 10 the above reductions. However, there is no means in PESs to equate the corresponding
 11 configurations.

12 The same problem can also be observed from a different angle. In order for an
 13 equivalence relation to satisfy the adequacy requirement, the corresponding equivalence
 14 classes of reductions must be partially ordered so as to form finitary prime algebraic
 15 (or equivalently, distributive) complete lattices (see Definition 55), since the domains
 16 of configurations of PESs ordered by inclusion are such lattices [36, 37]. However, it is
 17 well known that Lévy’s reduction space—permutation equivalence classes of co-initial
 18 finite reductions ordered by Lévy’s embedding relation \triangleleft (which generates permuta-
 19 tion equivalence)—is not, in general, a lattice but only an upper semi-lattice, as the
 20 greatest lower bound of two co-initial finite reductions w.r.t. \triangleleft need not exist [27].²
 21 As a consequence, although the lattice properties can be restored by taking the ideal
 22 completion [4], the resulting domain need not be finitary prime algebraic even when
 23 the reduction relation is non-duplicating. For example, Lévy’s configuration domain of
 24 the K - Ω term above is not finitary as the K -step $Kx\Omega \xrightarrow{K} x$ is a finitary element but it
 25 dominates (w.r.t. \triangleleft) an infinite number of elements: reductions of the form Ω^m .

26 Several authors have proposed different ways to get round this problem. Laneve
 27 [24] introduced *distributive* equivalence on β -reductions which only allows permutation
 28 of steps that cannot erase or duplicate one another, and consequently all equivalent
 29 sequences have the same length. Similarly, Corradini et al. [8] base their interpretation
 30 on the equivalence relation generated by permutations of disjoint redexes only, in a
 31 general categorical model of rewriting. Kennaway et al. [16] do work with permutation
 32 equivalence for orthogonal term graph rewriting, but they only restrict themselves to
 33 *needed* events of a normalizable term to cope with problems with erasure. As needed
 34 redexes cannot be erased, all needed normalizing reductions are of the same length,
 35 and on needed reductions, the permutation and distributive equivalences coincide (since
 36 the system is non-duplicating). Clark and Kennaway [7] also work with permutation-
 37 equivalence and allow for (finitely) non-normalizable terms, but restrict themselves to
 38 *standard* reductions, and their ordering on standard reductions is a strong restriction
 39 of Lévy’s embedding relation. And finally, extending work of Boudol [5], Mellès
 [30] gives a construction of distributive domains from *external* reductions [14], in

² Recursive Program Schemes are an exception [4].

1 an axiomatic framework of rewriting. Other authors (e.g. [6, 34]) work with linear
rewrite/transition systems (no duplication, no erasure), and do not therefore encounter
3 the erasure problem.

In order to cope with the adequacy problem, we extend the PES model in two
5 different but equivalent ways: by axiomatizing permutation-equivalence on event con-
figurations, by directly constraining the class of all equivalence relations on configura-
7 tions; and by axiomatizing the *inessentiality* or *erasure* relation expressing redundancy
of an event for a configuration. The latter extends by further axioms an incomplete
9 axiomatization of the inessentiality relation in [11]. In the resulting extended models,
deterministic permutation and *erasure* event structures, DPESs and DEESs, we can
11 identify finite configurations with different numbers of events but still representing the
same permutation-equivalence class of finite reductions. We show that Lévy’s configu-
13 ration domain—appropriately ordered permutation-equivalence classes of finite configu-
rations in a DEES/DPES—is isomorphic to Lévy’s reduction space of the corresponding
15 AZDFS, and therefore DPESs/DEESs give a *fully adequate* semantics to AZDFSs.

As a consequence, Lévy’s configuration domains in DEESs/DPESs need not be fini-
17 tary prime algebraic. However, having axiomatized the erasure relation and permutation
equivalence on event configurations, we can define orderings on configurations reflect-
19 ing the growth of information relative to *stable* sets of ‘results’, called *stable orderings*,
and thereby reconstruct finitary prime algebraic domains from configuration domains
21 of DEESs/DPESs: The isomorphism between AZDFSs and DEESs/DPESs discussed
above induces a concept of stable sets of configurations in DEESs/DPESs, and the
23 theory of normalization by neededness relative to stable sets of terms in AZDFSs [11]
has its counterpart in DEESs/DPESs; and this relativized concept of neededness en-
25 ables us to show that the stable orderings on configuration domains of DEESs/DPESs
do form finitary prime algebraic domains.

27 The paper is organized as follows. In Section 2, we review the existing theory of
DRSs and DFSs used in this paper. In Section 3 we introduce DPESs and DEESs
29 and establish their equivalence. In Section 4 we relate the latter event models with
affine SDRSs (ASDRSs) and AZDFSs, showing the equivalence of the four models.
31 In Section 5, we define Lévy’s configuration domain in a DPES/DEES and show its
isomorphism with Lévy’s reduction space of the corresponding ASDRS. We study nor-
33 malization by neededness in DEESs in Section 6, and use it to construct finitary prime
algebraic domains from configurations in DEESs/DPESs in Section 7. Conclusions ap-
35 pear in Section 8.

2. Deterministic residual and family structures

37 In this section, we recall some basic theory of *deterministic residual* and *family*
structures (DRSs and DFSs), developed in [11, 17, 18, 20]. DRSs and DFSs are *ab-*
39 *stract reduction systems* (ARSs) with an axiomatized notion of residual. In DFSs,
in addition, Lévy’s concept of redex-family is axiomatized. Related abstract residual
models are studied in [12, 13, 35, 29].

1 Our definition of ARSs allows multiple transitions with the same source and target, unlike Klop’s definition [22].

3 **Definition 1.** An ARS is a triple $A = (Ter, Red, \rightarrow)$ where Ter is a set of *terms*, ranged
 4 over by t, s, o, e ; Red is a set of *redexes* (or *redex occurrences*), ranged over by u, v, w ;
 5 and $\rightarrow : Red \mapsto (Ter \times Ter)$ is a function such that for any $t \in Ter$ there is only a finite
 6 set of $u \in Red$ such that $\mapsto(u) = (t, s)$, written $t \xrightarrow{u} s$. This set will be known as the
 7 redexes of term t , where $u \subseteq t$ denotes that u is a member of the redexes of t and $U \subseteq t$
 8 denotes that U is a subset of the redexes. Note that \rightarrow is a *total* function, so one can
 9 identify u with the triple $t \xrightarrow{u} s$. A *reduction* is a sequence $t \xrightarrow{u_1} t_2 \xrightarrow{u_2} \dots$. Reductions
 10 are denoted by P, Q, N . We write $P : t \rightarrow s$ or $t \xrightarrow{P} s$ if P denotes a reduction (sequence)
 11 from t to s . If P is finite and its final term coincides with the initial term of Q , then
 12 $P + Q$ denotes the concatenation of P and Q . Finally, u also denotes the reduction that
 13 contracts u .

14 **Definition 2** (*Deterministic residual structure*, Glauert and Khasidashvili [11]). A
 15 *deterministic residual structure* (DRS) is a pair $\mathcal{R} = (A, /)$, where A is an ARS and/is
 16 a *residual relation* on redexes relating redexes in the source and target term of every
 17 reduction $t \xrightarrow{u} s \in A$, such that for $v \subseteq t$, the set v/u of *residuals of v under u* is a set
 18 of redexes of s ; a redex in s may be a residual of only one redex in t under u , and
 19 $u/u = \emptyset$. If v has more than one u -residual, then u *duplicates* v . If $v/u = \emptyset$, then u
 20 *erases* v . A redex of s which is not a residual of any $v \subseteq t$ under u is said to be *u -new*
 21 or *created* by u . The set u/P of residuals of u under any finite reduction P is defined
 22 by transitivity.

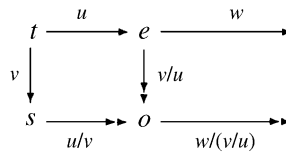
23 A *development* of $U \subseteq t$ is a reduction $P : t \rightarrow$ that only contracts residuals of redexes
 24 from U ; it is *complete* if it is finite and $U/P = \bigcup_{u \in U} u/P = \emptyset$. Development of \emptyset is
 25 identified with the empty reduction. U will also denote a complete development of
 26 $U \subseteq t$. The residual relation satisfies the following two axioms:

27 [FD] (*Finite developments* [12]) All developments are terminating; all complete de-
 28 velopments of $U \subseteq t$ end at the same term; and residuals of a redex $v \subseteq t$ under all
 29 complete developments of U are the same.

[weak acyclicity] [35] Let $u, v \subseteq t$, $u \neq v$, and $u/v = \emptyset$. Then $v/u \neq \emptyset$.³

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

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



³ This axiom is called [acyclicity] in [11], and is axiom (4) in [35].

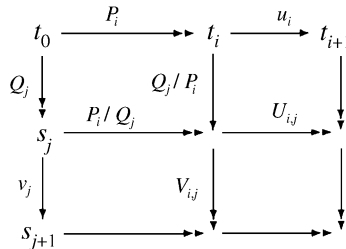
1 We call an SDRS *non-duplicating* or *affine*, ASDRS, if its residual relation is non-
 3 duplicating. Note that, since the only observables of DRSs are redexes, duplicating
 syntactic rewrite systems may still form ASDRSs. For example, the DRS corresponding
 5 to innermost reductions in an orthogonal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

In a DRS \mathcal{R} , the residual relation on redexes extends to all co-initial reductions
 7 exactly as in syntactic orthogonal rewrite systems [14, 26, 27, 35]: $(P_1 + P_2)/Q = P_1/Q +$
 $P_2/(Q/P_1)$ and $P/(Q_1 + Q_2) = (P/Q_1)/Q_2$, and *Lévy-equivalence* or *permutation-*
 9 *equivalence* is defined as the smallest relation on co-initial reductions satisfying: $U +$
 $V/U \approx V + U/V$ and $Q \approx Q' \Rightarrow P + Q + N \approx P + Q' + N$, where U and V are complete
 11 developments of redex sets in the same term. Further, one defines $P \trianglelefteq Q$ iff $P/Q = \emptyset$,
 and can show that $P \approx Q$ iff $P \trianglelefteq Q$ and $Q \trianglelefteq P$; and $P \trianglelefteq Q$ iff $Q \approx P + N$ for some N .
 13 Intuitively, $P \trianglelefteq Q$ expresses that Q does more work than P , and Q/P is the part of
 Q that remains from it after P . Finally, one shows that for any co-initial reductions
 15 P, Q , $P \sqcup Q \approx Q \sqcup P$, where $P \sqcup Q = P + Q/P$, and is the greatest lower bound of P and
 Q in the reduction space ordered by Lévy’s embedding relation \trianglelefteq . The above relations
 17 can equivalently be defined also using Klop’s method of commutative diagrams [21, 3].
 Finally, we will need the following *Cube Lemma*: for any finite co-initial reductions
 19 P, Q and N , $N/(P \sqcup Q) = N/(Q \sqcup P)$.

The following two lemmas, established in [11, 17], are fundamental in proving prop-
 21 erties in SDRSs. They extend [weak acyclicity] and [stability] axioms from one step
 reductions to any co-initial *external* reductions, that is, reductions that do not contract
 23 redexes having common residuals.

Definition 3.

- 25 • Let $u \in U \subseteq t$ and $P: t \rightarrow o$. We call P *external* to U (resp. u) if P does not contract
 residuals of redexes in U (resp. residuals of u).
- 27 • Let $P: t_0 \xrightarrow{P_i} t_i \xrightarrow{u_i} t_{i+1}$ and $Q: t_0 = s_0 \xrightarrow{Q_j} s_j \xrightarrow{v_j} s_{j+1} \rightarrow s_m$. 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 , $U_{i,j} \cap V_{i,j} = \emptyset$.



29

Lemma 4 (Stability). Let $P: t \rightarrow s$ be external to $Q: t \rightarrow e$, in a stable DRS, and let
 31 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 .

1 **Lemma 5** (Weak acyclicity). *Let P, Q be co-initial finite reductions in a DRS, and let P be external to Q . Then $Q \not\approx P$.*

3 Next, we recall some definitions and results from [17] concerning standardization of reductions in ASDRSs, used in this paper.

5 **Definition 6.**

• Let $P : t \rightarrow o$ and $u \subseteq t$, 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 $Q \approx P$ that is external to u , and call it P -*unnneeded* otherwise. We call u P -*essential* if there is no $Q \approx P$ 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 $Q : t \rightarrow o$, $P : t \xrightarrow{P'} s \rightarrow e$, and $u \subseteq s$. 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* 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 $P \approx Q$, by the Cube Lemma. The *discards* concept however does depend on the particular reduction in the permutation-equivalence class.

21 **Lemma 7.** *Let $P : s \rightarrow t \xrightarrow{u} e \rightarrow o$ in an ASDRS:*

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

23 (2) If $P : t \rightarrow s' \xrightarrow{w} o$, then $w \in s'$ is P -*needed*.

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

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

27 Self-needed reductions play the role of *standard* reductions in SDRSs, and the following algorithm is a standardization procedure for reductions in ASDRSs. Let $P : t \rightarrow s$.
 29 The *canonical standard variant* of P , $ST(P)$, is defined as follows: If $P = \emptyset$, then $ST(P) = \emptyset$. Otherwise, let $v \subseteq t$ 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 7). Then $ST(P) = v + ST(P/v)$. We write $Q \in STV(P)$ if
 31 $Q \in STA$ and $Q \approx P$, where STA denotes the set of all standard reduction, and call Q
 33 a *standard variant* of P . In particular, $ST(P) \in STV(P)$.

35 **Theorem 8** (Standardization). *For any finite reduction P in a stable non-duplicating DRS, $ST(P)$ is a standard reduction Lévy-equivalent to P .*

37 It is shown in [18, 20] that all standard variants of a finite reduction P can be constructed effectively.

1 **Definition 9** (*Deterministic family structure, Glauert and Khasidashvili [11]*). A *deterministic family structure* (DFS) is a triple $\mathcal{F} = (\mathcal{R}, \simeq, \hookrightarrow)$, where \mathcal{R} is a DRS;
 3 \simeq is an equivalence relation on redexes with *histories*; and \hookrightarrow is the *contribution* relation on co-initial families, defined as follows:

5 (1) For any co-initial finite 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 $P \trianglelefteq Q$, i.e., $P + Q/P \approx Q$, and
 7 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
 9 among redexes with histories containing \simeq_z . A *family* is an equivalence class of
 the family relation; families are ranged over by ϕ, ψ, \dots . $Fam(\)$ denotes the
 11 family of its argument.

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

13 [initial] Let $u, v \subseteq t$ 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 .

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

[creation] Let $e \xrightarrow{P} t \xrightarrow{u} s$, and let u create $v \subseteq s$. Then $Fam(Pu) \hookrightarrow Fam((P + u)v)$.

17 [FFD] (*Finite family developments*) Any reduction that contracts redexes of a finite
 number of families is terminating.⁴

19 Note that [contribution] can be viewed as a definition of \hookrightarrow rather than as an axiom.
 It is shown in [11] that every DFS is a stable DRS.

21 *Convention:* In this paper we only consider *comma-DRSs* and *comma-DFSs*. That
 is, the term set of any DRS will be the *reduction graph* of a term, called the *initial*
 23 term, i.e., the set of terms to which the initial term is reducible. The initial term will
 often be denoted by t_\emptyset , by analogy with the initial state \emptyset in an event structure. Further,
 25 in DFSs, families will always be considered relative to t_\emptyset , i.e., all histories start with
 t_\emptyset . Unless otherwise stated, by a *reduction* we mean a *finite* reduction. Reductions
 starting from t_\emptyset will also be called *initial* reductions.
 27

Definition 10.

29 • We call a DFS \mathcal{F} a *zig-zag* DFS, ZDFS, if its family relation is the zig-zag \simeq_z .
 Affine ZDFSs will be called AZDFSs.

31 • We call an affine DFS *separable* 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 $Fam((P +$
 33 $v)w') \neq Fam((P + v)w'')$.

The following lemma from [18, 20] gives a useful characterization of \trianglelefteq and \approx in
 35 ASDRSs via zig-zag classes. Below, $FAM(P)$ (resp. $SFAM(P)$) denotes the set of
 37 *zig-zag classes* whose member (resp. P -needed) redexes are contracted in P , in an
 ASDRS.

⁴This axiom is called [termination] in [11].

1 **Proposition 11.** *Let P and Q be initial reductions in an AZDFS. Then $P \trianglelefteq Q$ iff*
 2 *$\exists Q' \approx Q: FAM(P) \subseteq FAM(Q')$, and $P \approx Q$ iff $SFAM(P) = SFAM(Q)$. Further,*
 3 *$SFAM(P) = FAM(ST(P)) = \bigcap_{N \approx P} FAM(N)$. Consequently, if $P, Q \in STA$, then $P \approx Q$*
 4 *iff $FAM(P) = FAM(Q)$ (since in that case $P = ST(P)$ and $Q = ST(Q)$).*

5 The results of this paper are based on the following properties of ASDRSs, which
 6 show that ASDRSs and AZDFSs are essentially the same transition models, and that
 7 every zig-zag family has a unique (up to an equivalence on histories) minimal ele-
 8 ment [18, 20].

9 **Theorem 12** (Affine families). *Let \mathcal{R} be an ASDRS. Then $\mathcal{F}_R = (\mathcal{R}, \simeq_z, \hookrightarrow_z)$ is an*
 10 *affine zig-zag DFS, called the corresponding ZDFS of \mathcal{R} . Further, \mathcal{F}_R is the only*
 11 *separable DFS with R as the underlying SDRS, and in general, an affine DFS is*
 12 *separable iff it is a zig-zag DFS.*

13 **Theorem 13** (Affine extraction). *Every redex Pv in an ASDRS \mathcal{R} has a zig-zag re-*
 14 *lated redex $P'v'$ in canonical form, meaning that $P' \in STA$, its last step creates v' , and*
 15 *for any $P''v'' \simeq_z Pv$, $\exists N': P'' \approx P' + N' \wedge v'' = v'/N$. For any other canonical form*
 16 *P^*v^* of Pv , $P' \approx P^*$ and $v' = v^*$. Further, P' contracts exactly one redex in every con-*
 17 *tributor zig-zag family of Pv , in the corresponding AZDFS $\mathcal{F}_R = (\mathcal{R}, \simeq_z, \hookrightarrow_z)$ of \mathcal{R} .*

3. Deterministic permutation and erasure event structures

18 In this section, we introduce deterministic permutation and erasure event structures,
 19 DPESs and DEESs, which extend conflict-free prime event structures by an axiomatized
 20 permutation-equivalence and erasure relations, respectively, and establish an equiva-
 21 lence between them.

22 The purpose of this extension is to capture *directly* the phenomenon of *erasure* in
 23 event/poset models. The phenomenon is typical for languages based on rewrite sys-
 24 tems, but is not confined to them. In higher-order process calculi (e.g. [33]) processes
 25 (that can fire events) can be passed around as messages and therefore erased. Event
 26 structures are linear in nature and consequently the configuration domains enjoy very
 27 nice lattice properties; therefore event structures have been very successful in modelling
 28 *linear* process calculi where no actions can be erased or duplicated (see e.g. [6, 37]),
 29 such as CCS [31]. However, any attempt to adequately interpret higher-order process
 30 languages in event structures would face the same erasure problems discussed in the
 31 introduction. As in the case of reductions, the aim of axiomatizing permutation equiv-
 32 alence on event configurations is to be able to equate configurations which represent
 33 the same concurrent computation and differ only by irrelevant or *inessential* events.
 34 We believe that any equivalence relation on configurations fully capturing concurrency
 35 should contain the permutation-equivalence.

36 A *prime event structure* (PES) [36] is a triple $\mathcal{E} = (E, Con, \leq)$, where E is a set of
 37 *events*, ranged over by e, e_1, \dots ; the *consistency predicate* Con is a non-empty set of

1 finite subsets of E , denoted by X, Y, \dots ; and the *causal dependency relation* \leq is a partial order on E , such that $\{e\} \in \text{Con}$, $Y \subseteq X \in \text{Con} \Rightarrow Y \in \text{Con}$, $X \in \text{Con} \wedge \exists e' \in X. e \leq$
 3 $e' \Rightarrow X \cup \{e\} \in \text{Con}$, and $\{e' \mid e' \leq e\}$ is finite for any $e \in E$.

We only consider *conflict-free* or *deterministic* PESs, where no event can prevent
 5 others from occurring, and therefore the consistency predicate is the set of all finite subsets of E and will be omitted. Finite *configurations* of \mathcal{E} are finite *left-closed subsets*
 7 α, β, \dots of E , i.e., subsets $\mathcal{L}_{\text{fin}} = \{\alpha \subseteq_{\text{fin}} E \mid e \in \alpha \wedge e' < e \Rightarrow e' \in \alpha\}$. Except for Section 7, we only consider finite configurations, and will omit ‘finite’.

9 Below, for better readability, we write $\alpha + e$ for $\alpha \cup \{e\}$, and write $\alpha - e$ for $\alpha \setminus \{e\}$. Further, we define $[e]^\leq = \{e' \mid e' \leq e\}$, $[e]^{<} = \{e' \mid e' < e\}$, $[e]^\geq = \{e' \mid e \geq e'\}$ and
 11 $[e]^{>} = \{e' \mid e' > e\}$.

Definition 14. A *deterministic permutation event structure* (DPES) is a triple $\mathcal{P} = (E, \leq, \approx)$, where $\mathcal{E} = (E, \leq)$ is a conflict-free PES and $\approx \subseteq \mathcal{L}_{\text{fin}}(E) \times \mathcal{L}_{\text{fin}}(E)$ is an equivalence relation, called *Lévy-* or *permutation-equivalence*, satisfying the following axioms, where $\alpha, \beta, \gamma \in \mathcal{L}_{\text{fin}}(E)$:

- 13 [P0] $\forall e \in E: [e]^{<} \not\approx [e]^\leq$;
 14 [P1] $\alpha \approx \beta \wedge \alpha \cap \beta \subseteq \gamma \subseteq \alpha \cup \beta \Rightarrow \alpha \approx \gamma$;
 15 [P2] $\alpha \approx \beta \wedge \alpha + e, \beta + e \in \mathcal{L}_{\text{fin}}(E) \Rightarrow \alpha + e \approx \beta + e$;
 16 [P3] $\alpha \approx \alpha - e \wedge e < e' \wedge \alpha + e' \in \mathcal{L}_{\text{fin}}(E) \Rightarrow \alpha \approx \alpha + e'$;

DEESs extend *deterministic computation structures* [11] by further erasure axioms
 21 to enable a fully adequate treatment of erasure.

Definition 15. A *deterministic erasure event structure* (DEES) is a triple $\mathcal{C} = (E, \leq, \triangleright)$, where $\mathcal{E} = (E, \leq)$ is a conflict-free PES and $\triangleright \subseteq \mathcal{L}_{\text{fin}}(E) \times E$ is *inessentiality* or *erasure* relation (read $\alpha \triangleright e$ as: ‘ e is α -inessential’), satisfying the following axioms, where
 23 $\alpha, \beta \in \mathcal{L}_{\text{fin}}(E)$:

- 24 [E0] $\forall e \in E: \emptyset \not\triangleright e$;
 25 [E1] $\alpha \triangleright e \wedge \alpha \subseteq \beta \in \mathcal{L}_{\text{fin}}(E) \Rightarrow \beta \triangleright e$;
 26 [E2] $\alpha \triangleright e' \wedge \alpha \triangleright e \wedge \alpha - e' \in \mathcal{L}_{\text{fin}}(E) \Rightarrow \alpha - e' \triangleright e$;
 27 [E3] $\alpha \triangleright e \wedge e < e' \Rightarrow \alpha \triangleright e'$;
 28 [E4] $\alpha \cup [e]^{<} \triangleright e \Rightarrow \alpha \triangleright e$.

31 We may write $\alpha \triangleright X$ if $\forall e \in X: \alpha \triangleright e$. Sometimes we write $\mathcal{L}_{\text{fin}}(\mathcal{P})$ for $\mathcal{L}_{\text{fin}}(E)$ if E is the domain of a DPES \mathcal{P} , and similarly for $\mathcal{L}_{\text{fin}}(\mathcal{C})$.

33 Before explaining the intuition behind the axioms, we first define translations between DPESs and DEESs.

35 **Definition 16.** For a DPES $\mathcal{P} = (E, \leq, \approx)$, we define an *erasure relation* $\triangleright_{\approx} \subseteq \mathcal{L}_{\text{fin}}(E) \times E$ and the corresponding structure $\mathcal{C}_{\mathcal{P}} = (E, \leq, \triangleright_{\approx})$ as follows:

- 36 • $\alpha \triangleright_{\approx} e$ iff $\alpha \cup [e]^\leq \approx (\alpha \cup [e]^\leq) \setminus [e]^\geq$.
 37 In particular, when $e \in \alpha$ and $\alpha - e \in \mathcal{L}_{\text{fin}}(E)$,
 38 • $\alpha \triangleright_{\approx} e$ iff $\alpha \approx \alpha - e$.

1 **Definition 17.** For a DEES $\mathcal{C} = (E, \leq, \triangleright)$, we define an equivalence relation $\approx_{\triangleright} \subseteq \mathcal{L}_{\text{fin}}(E) \times \mathcal{L}_{\text{fin}}(E)$ and the corresponding structure $\mathcal{P}_{\mathcal{C}} = (E, \leq, \approx_{\triangleright})$ as follows:
 3 $\alpha \approx_{\triangleright} \beta$ iff $ST_{\triangleright}(\alpha) = ST_{\triangleright}(\beta)$, where $ST_{\triangleright}(\alpha) = \{e \in \alpha \mid \alpha \not\triangleright e\}$. (The subscript \triangleright in $ST_{\triangleright}(\alpha)$ will often be omitted.) $ST(\alpha)$ will be called the *standard variant* of α .

5 By looking at a configuration α as an initial reduction P_{α} firing redexes corresponding to events in α , and by interpreting $\alpha \triangleright e$ as ‘ Qv is P -inessential’, where Qv is a redex representing v (so Q contracts redexes corresponding to events on which e causally depends), it is not difficult to understand the intuition behind the erasure axioms. For
 7 example, [E0] says that an empty reduction cannot erase any redex, and [E1] says that if Qv is P_{α} -inessential and $P_{\alpha} \triangleleft P_{\beta}$, then Qv is P_{β} -inessential as well. This intuition
 9 will become more precise in the next section. Further, a close inspection of the above definitions shows that the DPES axiom [P0] is a counterpart of the DEES axioms
 11 [E0] and [E4], and similarly for [P3] and [E3]. The combination of [P1] and [P2] have the same effect as that of [E1] and [E2]. Axiom [E4] does not have a ‘direct’
 13 counterpart among DPES axioms, but it corresponds to the fact that, according to Definition 6, a redex with history Pv is Q -inessential iff v is Q/P -inessential. Similarly,
 15 [P0] corresponds to Lemma 7(2).
 17

In order to establish equivalence of DEESs and DPESs, we need a few lemmas.

19 **Lemma 18.** Let $\alpha \in \mathcal{L}_{\text{fin}}(\mathcal{C})$, in a DEES \mathcal{C} . Then $ST(\alpha) \in \mathcal{L}_{\text{fin}}(\mathcal{C})$. Further, $ST(ST(\alpha)) = ST(\alpha)$, hence $\alpha \approx_{\triangleright} ST(\alpha)$.

21 **Proof.** Let $e \in ST(\alpha) \wedge e' < e$. Then $e \in \alpha$ and $\alpha \not\triangleright e$, implying by [E3] that $\alpha \not\triangleright e'$. Since $\alpha \in \mathcal{L}_{\text{fin}}(\mathcal{C})$ and $e' < e$, $e' \in \alpha$, implying $e' \in ST(\alpha)$. Hence $ST(\alpha) \in \mathcal{L}_{\text{fin}}(\mathcal{C})$. If, on
 23 the contrary, $ST(ST(\alpha)) \neq ST(\alpha)$, i.e., $\exists e'' \in ST(\alpha): ST(\alpha) \triangleright e''$, then $\alpha \triangleright e''$ by [E1], contradicting $e'' \in ST(\alpha)$. Now $\alpha \approx_{\triangleright} ST(\alpha)$ follows immediately from Definition 17.
 25 \square

Lemma 19. Let $\alpha \in \mathcal{L}_{\text{fin}}(\mathcal{C})$, in a DEES \mathcal{C} . Then $\alpha \triangleright e$ iff $ST(\alpha) \triangleright e$.

27 **Proof.** (\Rightarrow) Let $\alpha = ST(\alpha) \cup \{e_1, \dots, e_n\}$, where the enumeration e_1, \dots, e_n respects the order on E , let $\alpha_0 = \alpha$, and let $\alpha_{i+1} = \alpha_i + e_{i+1}$ (hence $\alpha = \alpha_n$). Denote $e = e_0$. By the
 29 definition of $ST(\alpha)$ and the assumption, we have $\alpha_n \triangleright e_i$ for all $i = 0, 1, \dots, n$. By [E2], we have from $\alpha_n \triangleright e_n$ and $\alpha_n \triangleright e_i$ that $\alpha_{n-1} \triangleright e_i$. Hence, again by [E2], $\alpha_{n-2} \triangleright e_i$, and
 31 so on. Finally, we get $\alpha_0 \triangleright e_i$ for all i , and in particular, $\alpha_0 \triangleright e$.

(\Leftarrow) By [E1]. \square

33 **Lemma 20.** Let $\alpha, \beta \in \mathcal{L}_{\text{fin}}(E)$ and $\alpha \subseteq \beta$, in a DEES \mathcal{C} . Then $ST(\alpha) = ST(\beta)$ iff $\beta \triangleright \beta \setminus \alpha$.

35 **Proof.** (\Rightarrow) By Definition 17, since $\beta \setminus \alpha \subseteq \beta \setminus ST(\alpha)$.

(\Leftarrow) Since $\beta \triangleright \beta \setminus \alpha$, $ST(\beta) \subseteq \alpha$. Let $e' \in ST(\beta)$. Then $e' \in \beta$ and $\beta \not\triangleright e'$, implying by [E1] and $\alpha \subseteq \beta$ that $\alpha \not\triangleright e'$, i.e., $e' \in ST(\alpha)$. Hence $ST(\beta) \subseteq ST(\alpha)$. For the converse,

1 take $e'' \in ST(\alpha)$. Then $e'' \in \alpha$ and $\alpha \not\triangleright e''$. Therefore, $e'' \in \beta$ and by [E1] $ST(\beta) \not\triangleright e''$,
 implying by Lemma 19 that $\beta \triangleright e''$, i.e., $e'' \in ST(\beta)$. Hence $ST(\alpha) = ST(\beta)$. \square

3 **Proposition 21.** *Let $\alpha, \beta, \gamma \in \mathcal{L}_{\text{fin}}(\mathcal{C})$, in a DEES \mathcal{C} , and let $\alpha \cap \beta \subseteq \gamma \subseteq \alpha \cup \beta$. Then $ST(\alpha) = ST(\beta) \Leftrightarrow \gamma \triangleright \alpha \cup \beta \setminus \alpha \cap \beta$. Consequently, $\alpha \approx_{\triangleright} \beta$ iff $\gamma \triangleright \alpha \cup \beta \setminus \alpha \cap \beta$.*

5 **Proof.** (\Rightarrow) $ST(\alpha) = ST(\beta) \Rightarrow ST(\alpha) \subseteq \alpha \cap \beta$, and by the definition of $ST(\alpha)$, Lemma 19,
 and [E1], $ST(\alpha) = ST(\beta) \Rightarrow ST(\alpha) \triangleright \beta \setminus \alpha \cup \alpha \setminus \beta \Rightarrow \gamma \triangleright \alpha \cup \beta \setminus \alpha \cap \beta$.

7 (\Leftarrow) By [E1] and Lemma 20, $\alpha \cap \beta \triangleright \alpha \cup \beta \setminus \alpha \cap \beta \Rightarrow \alpha \triangleright \alpha \setminus \beta \wedge \beta \triangleright \beta \setminus \alpha \Rightarrow ST(\alpha) =$
 $ST(\alpha \cap \beta) = ST(\beta)$. \square

9 **Theorem 22.** *For any DEES $\mathcal{C} = (E, \leq, \triangleright)$, $\mathcal{P}_{\mathcal{C}} = (E, \leq, \approx_{\triangleright})$ is a DPES.*

Proof. We need to show that \approx_{\triangleright} satisfies the permutation axioms [P0]–[P3].

11 [P0] Let $e \in E$. Note that $[e]^{\leq} \triangleright e$ would imply by [E2] that $[e]^{\leq} \triangleright e$, implying $\emptyset \triangleright e$
 by [E4], contradicting [E0]. Hence $[e]^{\leq} \not\triangleright e$, i.e., $e \in ST([e]^{\leq})$, and therefore
 13 $ST([e]^{\leq}) \neq ST([e]^{\leq})$. Thus $[e]^{\leq} \not\approx_{\triangleright} [e]^{\leq}$ by Definition 17.

[P1] Let $\alpha \approx_{\triangleright} \beta$, $\alpha \cap \beta \subseteq \gamma \subseteq \alpha \cup \beta$, and $\gamma \in \mathcal{L}_{\text{fin}}(E)$. By Definition 17, $ST(\alpha) = ST(\beta)$,
 15 hence $ST(\alpha) \subseteq \alpha \cap \beta \subseteq \gamma$. Further, $\forall e \in \alpha \cup \beta \setminus ST(\alpha)$, $\alpha \triangleright e$ or $\beta \triangleright e$, hence $ST(\alpha) \triangleright e$
 by Lemma 19. Hence $ST(ST(\alpha)) = ST(\gamma)$ by Proposition 21, and $ST(\alpha) = ST(\gamma)$
 17 by Lemma 18, implying $\gamma \approx_{\triangleright} \alpha$ by Definition 17.

[P2] Let $\alpha \approx_{\triangleright} \beta$ and $\alpha + e, \beta + e \in \mathcal{L}_{\text{fin}}(E)$. By Definition 17, $ST(\alpha) = ST(\beta)$.
 19 Assume first that $ST(\alpha) \triangleright e$. Then $\alpha + e \triangleright e$ by [E1] and $ST(\alpha + e) = ST(\alpha)$
 by Proposition 21. Similarly, $ST(\beta + e) = ST(\beta)$. Hence, $ST(\alpha + e) = ST(\beta + e)$,
 21 implying by Definition 17 that $\alpha + e \approx_{\triangleright} \beta + e$. Now let $ST(\alpha) \not\triangleright e$. Then, by
 Lemma 19 and [E3], $\forall e' \in \alpha \setminus ST(\alpha)$: $e' \not\triangleright e$. Thus $\alpha + e \in \mathcal{L}_{\text{fin}}(E) \Rightarrow ST(\alpha) + e \in$
 23 $\mathcal{L}_{\text{fin}}(E)$. By Lemma 19, $ST(\alpha) \not\triangleright e$ implies $\alpha \not\triangleright e$, thus $e \in ST(\alpha)$ iff $e \in \alpha$, and
 therefore $\alpha + e \setminus (ST(\alpha) + e) = \alpha \setminus ST(\alpha)$. Hence $\alpha + e \triangleright \alpha + e \setminus (ST(\alpha) + e)$ by [E1]
 25 (as $\alpha \triangleright \alpha \setminus ST(\alpha)$), and we have by Proposition 21 that $ST(\alpha + e) = ST(ST(\alpha) + e)$,
 thus $\alpha + e \approx_{\triangleright} ST(\alpha) + e$ by Definition 17. Similarly, $\beta + e \approx_{\triangleright} ST(\beta) + e$, implying
 27 $\alpha + e \approx_{\triangleright} \beta + e$ (since $ST(\alpha) = ST(\beta)$).

[P3] Let $\alpha \approx_{\triangleright} \alpha - e$, $e < e'$ and $\alpha + e' \in \mathcal{L}_{\text{fin}}(E)$ (hence $e \in \alpha$). By Definition 17, $ST(\alpha) =$
 29 $ST(\alpha - e)$, implying $\alpha \triangleright e$. Hence, by [E3], $\alpha \triangleright e'$, implying by [E1] that $\alpha + e' \triangleright e'$.
 Hence, by Proposition 21, $ST(\alpha + e') = ST(\alpha)$, thus $\alpha + e' \approx_{\triangleright} \alpha$ by Definition 17.

31 \square

Theorem 23. *For any DPES $\mathcal{P} = (E, \leq, \approx)$, $\mathcal{C}_{\mathcal{P}} = (E, \leq, \triangleright_{\approx})$ is a DEES.*

33 **Proof.** We need to show that \triangleright_{\approx} satisfies the erasure axioms [E0]–[E4].

[E0] By Definition 16 and [P0].

35 [E3] Let $\alpha \triangleright_{\approx} e$ and $e < e'$. We want to show that $\alpha \triangleright_{\approx} e'$. We can assume that
 37 $\not\exists e''$: $e < e'' < e'$. Assume first that $e, e' \in \alpha$ (hence $\alpha \cup [e]^{\leq} = \alpha \cup [e']^{\leq} = \alpha$). By
 Definition 16, $\alpha \triangleright_{\approx} e \Rightarrow \alpha \approx \alpha \setminus [e]^{\geq}$. But $\alpha \setminus [e]^{\geq} \subseteq \alpha \setminus [e']^{\geq} \subseteq \alpha$, implying by [P1]

- 1 that $\alpha \approx \alpha \setminus [e']^\geq$, i.e., by Definition 16, $\alpha \triangleright_{\approx} e'$. Now let $e \in \alpha$ and $e' \notin \alpha$. Let
 $\gamma = \alpha \setminus [e]^\geq$. By Definition 16, $\alpha \triangleright_{\approx} e$ iff $\gamma \triangleright_{\approx} e$ iff $\gamma \approx \gamma - e$. Further, $\gamma \cup [e']^\leq = \gamma$
 3 $+ e'$ and $\gamma \cup [e']^\leq \setminus [e']^\geq = \gamma$. By [P3], $\gamma \approx \gamma + e'$, implying $\gamma \triangleright_{\approx} e'$ and equiva-
 5 lently $\alpha \triangleright_{\approx} e'$ by Definition 16. Finally, let $e, e' \notin \alpha$. Take $\beta = \alpha \cup [e]^\leq$. Then, by
 Definition 16, $\alpha \triangleright_{\approx} e$ iff $\beta \triangleright_{\approx} e$ and $\alpha \triangleright_{\approx} e'$ iff $\beta \triangleright_{\approx} e'$, and we have $e \in \beta$ and
 $e' \notin \beta$ like in the previous case, and [E3] follows.
- 7 [E1] Let $\alpha \triangleright_{\approx} e \wedge \alpha \subseteq \beta \in \mathcal{L}_{\text{fin}}(E)$. We can assume that $\beta = \alpha + e'$ and $e' \notin \alpha$. If $e \geq e'$,
 then $\alpha + e' \cup [e]^\leq = \alpha \cup [e]^\leq$ and $\alpha + e' \cup [e]^\leq \setminus [e]^\geq = \alpha \cup [e]^\leq \setminus [e]^\geq$, thus $\alpha \triangleright_{\approx} e$
 9 iff $\alpha + e' \triangleright_{\approx} e$ by Definition 16. If $e \not\geq e'$ and $e' \not\leq e$, then $\alpha + e' \cup [e]^\leq = \alpha \cup [e]^\leq + e'$
 and $\alpha + e' \cup [e]^\leq \setminus [e]^\geq = \alpha \cup [e]^\leq \setminus [e]^\geq + e'$, and $\alpha + e' \triangleright_{\approx} e$ follows from $\alpha \triangleright_{\approx} e$
 11 by [P2] and Definition 16. Finally, if $e < e'$, then $\alpha + e' \in \mathcal{L}_{\text{fin}}(E) \Rightarrow e \in \alpha$. By
 [E3] which we have already proven above for \triangleright_{\approx} , $\alpha \triangleright_{\approx} e \Rightarrow \alpha \triangleright_{\approx} e'$; thus by
 13 Definition 16 we have from $\alpha \triangleright_{\approx} e$ that $\alpha = \alpha \cup [e]^\leq \approx \alpha \cup [e]^\leq \setminus [e]^\geq$, and we
 have from $\alpha \triangleright_{\approx} e'$ that $\alpha \approx \alpha + e'$. Hence $\alpha + e' \cup [e]^\leq \setminus [e]^\geq \approx$ (since $e < e'$)
 15 $\approx \alpha \cup [e]^\leq \setminus [e]^\geq \approx \alpha \approx \alpha + e' = \alpha + e' \cup [e]^\leq$, which means by Definition 16 that
 $\alpha + e' \triangleright e$.
- 17 [E2] We can assume that $e' \in \alpha$. Let $\alpha \triangleright_{\approx} e'$, $\alpha \triangleright_{\approx} e$ and $\alpha - e' \in \mathcal{L}_{\text{fin}}(E)$. By
 Definition 16, $\alpha \triangleright_{\approx} e \Rightarrow \alpha \cup [e]^\leq \approx \alpha \cup [e]^\leq \setminus [e]^\geq$ and $\alpha \triangleright_{\approx} e' \Rightarrow \alpha \approx \alpha - e'$, im-
 19 plying by repeated application of [P2] that $\alpha \cup [e]^\leq \approx \alpha - e' \cup [e]^\leq$. It is easy to
 check that $(\alpha \cup [e]^\leq \setminus [e]^\geq) \cap (\alpha - e' \cup [e]^\leq) = (\alpha \cup [e]^\leq) \cap (\alpha - e' \cup [e]^\leq) \setminus [e]^\geq =$
 21 $\alpha - e' \cup [e]^\leq \setminus [e]^\geq$ implying by [P1] that $\alpha - e' \cup [e]^\leq \approx \alpha - e' \cup [e]^\leq \setminus [e]^\geq$.
 Hence, $\alpha - e' \triangleright_{\approx} e$ by Definition 16.
- 23 [E4] We need to show that $\alpha \cup [e]^\leq \triangleright_{\approx} e \Rightarrow \alpha \triangleright_{\approx} e$. Indeed, by Definition 16, $\alpha \cup [e]^\leq$
 $\triangleright e$ iff $(\alpha \cup [e]^\leq) \cup [e]^\leq \setminus [e]^\geq \approx (\alpha \cup [e]^\leq) \cup [e]^\leq \Leftrightarrow \alpha \cup [e]^\leq \setminus [e]^\geq \approx \alpha \cup [e]^\leq$
 25 iff $\alpha \triangleright_{\approx} e$. \square

Lemma 24. *Let $\mathcal{P} = (E, \leq, \approx)$ be a DPES, and let $\alpha \in \mathcal{L}_{\text{fin}}(E)$. Then $\alpha \approx ST_{\triangleright_{\approx}}(\alpha)$.*

- 27 **Proof.** By Theorem 23 and Lemma 18, $ST_{\triangleright_{\approx}}(\alpha)$ is a configuration. Let $\alpha \setminus ST_{\triangleright_{\approx}}(\alpha) =$
 $\{e_1, \dots, e_n\}$. By Definition 16, $\alpha \triangleright_{\approx} e_i \Rightarrow \alpha \approx \alpha \setminus [e_i]^\geq$. Clearly, $\bigcap_{i=1}^n \alpha \setminus [e_i]^\geq \subseteq ST_{\triangleright_{\approx}}$
 29 $(\alpha) \subseteq \alpha$, implying by [P1] and $\alpha \approx \alpha \setminus [e_i]^\geq$ that $\alpha \approx ST_{\triangleright_{\approx}}(\alpha)$. \square

31 The next proposition gives a characterization of \approx via \triangleright_{\approx} , needed for establishing
 an isomorphism between DPESs and DEESs.

Proposition 25. *In a DPES $\mathcal{P} = (E, \leq, \approx)$, $\alpha \approx \beta$ iff $ST_{\triangleright_{\approx}}(\alpha) = ST_{\triangleright_{\approx}}(\beta)$.*

- 33 **Proof.** (\Leftarrow) Immediate from Lemma 24.

(\Rightarrow) By Lemma 24, $\alpha \approx ST_{\triangleright_{\approx}}(\alpha) \approx ST_{\triangleright_{\approx}}(\beta)$. Suppose on the contrary that $ST_{\triangleright_{\approx}}(\alpha)$
 35 $\neq ST_{\triangleright_{\approx}}(\beta)$, say $\exists e \in ST_{\triangleright_{\approx}}(\alpha) \setminus (ST_{\triangleright_{\approx}}(\alpha) \cap ST_{\triangleright_{\approx}}(\beta))$. By [P1], $ST_{\triangleright_{\approx}}(\alpha) \approx ST_{\triangleright_{\approx}}(\alpha) \cap$
 $ST_{\triangleright_{\approx}}(\beta)$. Further, $ST_{\triangleright_{\approx}}(\alpha) \cap ST_{\triangleright_{\approx}}(\beta) \subseteq \alpha \setminus [e]^\geq \subseteq \alpha$ (since $ST_{\triangleright_{\approx}}(\alpha) \cap ST_{\triangleright_{\approx}}(\beta)$ is left-
 37 closed), hence again by [P1] $\alpha \cup [e]^\leq \setminus [e]^\geq = \alpha \setminus [e]^\geq \approx \alpha = \alpha \cup [e]^\leq$, implying by
 Definition 16 that $\alpha \triangleright_{\approx} e$, contradicting $e \in ST_{\triangleright_{\approx}}(\alpha)$. \square

- 1 **Theorem 26** (Equivalence of DPESs and DEESs). (1) For any DEES $\mathcal{C} = (E, \leq, \triangleright)$,
 $\mathcal{C}_{P_C} = \mathcal{C}$.
 3 (2) For any DPES $\mathcal{P} = (E, \leq, \approx)$, $\mathcal{P}_{C_P} = \mathcal{P}$.

Proof. Note that the translations of DEESs into DPES and vice versa does not effect
 5 the sets of events and the causality relation. Thus $\mathcal{C}_{P_C} = (E, \leq, \triangleright_{\approx_{\triangleright}})$ and $\mathcal{P}_{C_P} = (E, \leq,$
 $\approx_{\triangleright_{\approx}})$. We need to show that (1) $\triangleright = \triangleright_{\approx_{\triangleright}}$ and (2) $\approx = \approx_{\triangleright_{\approx}}$.

- 7 (1) We need to prove that $\alpha \triangleright e$ in \mathcal{C} iff $\alpha \triangleright_{\approx_{\triangleright}} e$ in \mathcal{C}_{P_C} . By [E4], [E1] and [E3],
 $\alpha \triangleright e$ iff $\alpha \cup [e] \leq \triangleright e$ iff $\alpha \cup [e] \leq \triangleright [e] \triangleright$. By Proposition 21, the latter holds iff
 9 $ST_{\triangleright}(\alpha \cup [e] \leq) = ST_{\triangleright}(\alpha \cup [e] \leq \setminus [e] \triangleright)$, which, by Definition 17 means that $\alpha \cup$
 $[e] \leq \approx \triangleright \alpha \cup [e] \leq \setminus [e] \triangleright$, meaning $\alpha \triangleright_{\approx_{\triangleright}} e$ by Definition 16.
 11 (2) We need to prove that $\alpha \approx \beta$ in \mathcal{P} iff $\alpha \approx_{\triangleright_{\approx}} \beta$ in \mathcal{P}_{C_P} . By Proposition 25 and
 Definition 17, $\alpha \approx \beta$ iff $ST_{\triangleright_{\approx}}(\alpha) = ST_{\triangleright_{\approx}}(\beta)$ iff $\alpha \approx_{\triangleright_{\approx}} \beta$. \square

- 13 From now on, we will merge DEESs $\mathcal{C} = (E, \leq, \triangleright)$ and DPESs $\mathcal{P} = (E, \leq, \approx)$ into
 one model $(E, \leq, \triangleright, \approx)$, where (E, \leq) is a prime event structure, the erasure relation
 15 \triangleright satisfies axioms [E0]–[E4], and $\approx = \approx_{\triangleright}$. The obtained results allow us to assume
 that the permutation-equivalence \approx satisfies axioms [P0]–[P3], and that $\triangleright = \triangleright_{\approx}$. We
 17 call such models again DEESs, and denote by \mathcal{C} .

4. Relating conflict-free stable transition and event models

- 19 We now define translations between AZDFSs and DEESs and show that they com-
 mute, implying equivalence of the two computational models.

4.1. Translation of ASDRSs and AZDFSs into DEESs

Definition 27. With an ASDRS \mathcal{R} and its corresponding AZDFS $\mathcal{F} = \mathcal{F}_{\mathcal{R}} = (\mathcal{R},$
 23 $\simeq_z, \hookrightarrow_z)$, we associate the DEES $\mathcal{C}_{\mathcal{R}} = \mathcal{C}_{\mathcal{F}} = (E_{\mathcal{F}}, \leq, \triangleright, \approx)$, where

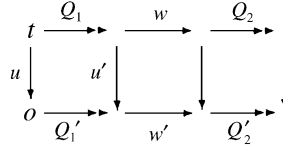
- $E_{\mathcal{F}} = FAM(\mathcal{F})$, the set of all families of \mathcal{F} ;
- 25 • $\phi < \psi$ iff $\phi \hookrightarrow_z \psi$;
- $\alpha \approx \beta$ iff $\forall \phi \in \alpha \cup \beta \setminus \alpha \cap \beta, \exists P : t_0 \rightarrow s : FAM(P) \subseteq \alpha \cap \beta$ and P discards a canonical
 27 element of ϕ (see Theorem 13);
- $\alpha \triangleright \phi$ iff $\exists P : t_0 \rightarrow s : FAM(P) \subseteq \alpha$ and P discards a canonical element of ϕ .

- 29 We need to check that $\mathcal{C}_{\mathcal{F}}$ in the above definition is indeed a DEES. We need a
 number of technical lemmas to achieve this.

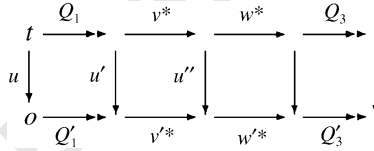
- 31 **Lemma 28.** Let $Q : t \rightarrow s, v \subseteq t, Q' \in STV(Q)$, and let v be Q -inessential, in an AS-
 DRS. Then Q' discards v .

- 33 **Proof.** Since v is Q -inessential, $v/Q = \emptyset$ and v is Q -unnecessary. Since Q' is Q -needed,
 it is external to v by Lemma 7(4), and $v/Q' = v/Q = \emptyset$, i.e., Q' discards v . \square

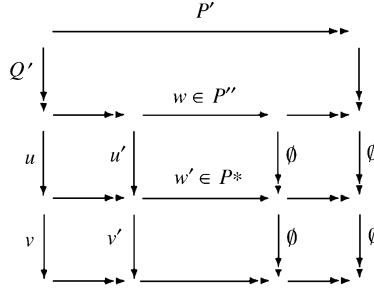
- 1 **Lemma 29.** Let $Q: t \rightarrow s$ be standard and let $t \xrightarrow{u} o$, in an ASDRS. Then Q/u is standard.
- 3 **Proof.** By induction on the length of Q . Let $Q' = Q/u$. Suppose on the contrary that Q' is not standard and let w' be the last Q' -unnecessary step in Q' , say $Q' = Q'_1 + w' + Q'_2$,
- 5 and let $Q = Q_1 + w + Q_2$, where $w' = w/u'$ and $u' = u/Q_1$ (see the figure).



- 7 By Lemma 7(2), w' is not the last step of Q' . So let $Q_2 = v + Q_3$ and $Q'_2 = v' + Q'_3$. By the choice of w' , v' is Q' -needed, and w' does not create v' by Lemma 7(3). Since
- 9 $w + Q_2$ is self-needed and $w' + Q'_2$ is not, $u' \neq \emptyset$. Hence, by [stability], w does not create v . Thus w' and v' , as well as w and v , can be permuted in $w' + v' + Q'_3$ and
- 11 $w + v + Q_3$, respectively, yielding $v'^* + w'^* + Q'_3$ and $v^* + w^* + Q_3$ (see the figure below). Then by Lemma 7(4), $w^* + Q_3$ is self-needed and $w'^* + Q'_3$ is not self-needed
- 13 (and therefore $u'' = u'/v^* \neq \emptyset$), contradicting the induction assumption. \square



- 15 **Lemma 30.** Let $Q: t \rightarrow s$, $P: t \rightarrow o$, $v \subseteq o$, $Q' \in STV(Q)$, and let Pv be Q -inessential, in an ASDRS. Then Q' discards Pv .
- 17 **Proof.** By Lemma 29, $Q'/P \in STV(Q/P)$. And since v is Q/P -inessential, Q'/P discards v by Lemma 28. \square
- 19 **Lemma 31.** Let $P: t \rightarrow s$, $Q: t \xrightarrow{Q'} o \xrightarrow{u} e \xrightarrow{v} s'$, let $Q'u$ be P -inessential, and let u create v , in an ASDRS. Then $(Q' + u)v$ is P -inessential too.
- 21 **Proof.** Let $P' = ST(P)$, $P'' = P'/Q'$, and $P^* = P''/u$. By Lemma 30, P'' discards u . So let w be the step of P'' that discards a residual u' of u , and let v' be the corresponding
- 23 residual of v along P^* , see the figure. If $v' \neq \emptyset$, then $v'/(w/u') = \emptyset$ by [stability] (since $u'/w = \emptyset$). So, in any case, $v/P^* = \emptyset$. By the Stability Lemma, P^* is external to v (since P'' is external to u). Hence P^* discards v , i.e., v is P -inessential. \square



1

Lemma 32. Let $P: t_0 \rightarrow s$, ϕ be a family in an AZDFS \mathcal{F} , let $Qv, Q'v' \in \phi$, and let Qv be canonical. If Qv is P -inessential, then so is $Q'v'$. Further, if $Q' \trianglelefteq P$, then the converse is also true.

Proof. By the Affine Extraction Theorem, there is Q'' such that $Q' \approx Q + Q''$ and $v' = v/Q''$. If Qv is P -inessential, then $ST(P)/Q$ discards v by Lemma 30, and $ST(P)/Q'$ discards v' by the Cube Lemma. Conversely, if $Q' \trianglelefteq P$, then $Q''/(P/Q) = \emptyset$, and $v'/(P/Q') = v/(P/Q) = \emptyset$ again by the Cube Lemma, thus Qv is P -inessential. \square

Lemma 33. Let $t_0 \xrightarrow{Q} s \xrightarrow{u} s'$, and $t_0 \xrightarrow{P} o \xrightarrow{v} o'$, and $Qu \simeq_z Pv$, in an AZDFS \mathcal{F} . Then $v/((Q + u)/P) = \emptyset$.

Proof. If $v' = v/(Q/P) = \emptyset$, then we are done. Otherwise, by the Affine Extraction Theorem, there are $Nw \simeq_z Qu$ and reductions N' and N'' such that $Q \approx N + N'$, $P \approx N + N''$, $u = w/N'$ and $v = w/N''$. Further, $N' + P/Q \approx N'' + Q/P$, and by the Cube Lemma, $v' = w/(N' + P/Q)$. But u is the only N' -residual of w , thus $v' = u/(P/Q)$, implying $v/((Q + u)/P) = \emptyset$. \square

Lemma 34. Let $P \in STA$, $FAM(P) = F_1 \cup F_2$, $F_1 \cap F_2 = \emptyset$ and $\forall \phi \in F_2, \forall \psi \in F_1 : \phi \not\leftrightarrow \psi$, in an AZDFS \mathcal{F} . Then $\exists Q \in STV(P) : Q = Q_1 + Q_2 \wedge FAM(Q_i) = F_i, i = 1, 2$.

Proof. Let u be the latest step in P such that $\phi = Fam(u) \in F_2$ and it has a next step v with $\psi = Fam(v) \in F_1$. Then $\phi \not\leftrightarrow \psi$ by the hypothesis, and u does not create v by [creation]. Hence u and v can be permuted, yielding another standard reduction by Lemma 7(4). The resulting reduction can be transformed similarly until all steps in F_2 are pushed to the end. The transformation terminates as it does not change the length of the reduction. \square

Theorem 35. For any AZDFS \mathcal{F} , \mathcal{C}_F is a DEES.

Proof. We need to show that $\mathcal{C}_F = (E_{\mathcal{F}}, \xrightarrow{\quad}, \triangleright, \approx)$ satisfies the erasure axioms [E0]–[E4], and that $\approx = \approx_{\triangleright}$.

- 1 [E0] Immediate from Definitions 6 and 27.
 [E1] Again from Definitions 6 and 27.
- 3 [E2] Let $\alpha \triangleright \phi$, $\alpha \triangleright \phi'$, and let $\alpha - \phi \in \mathcal{L}_{\text{fin}}(E)$. Then there are P and P' such that
 5 $FAM(P), FAM(P') \subseteq \alpha$, a canonical element Qv of ϕ is P -inessential, and a canonical
 7 element $Q'v'$ of ϕ' is P' -inessential. Let $P^* = P \sqcup P'$. Then $FAM(P^*) \subseteq \alpha$ and
 both Qv and $Q'v'$ are P^* -inessential. By Lemma 32, all members of ϕ and ϕ'
 are P^* -inessential. Since $ST(P^*)$ is P^* -essential, it does not contract members of
 9 ϕ and ϕ' . Hence $FAM(ST(P^*)) \subseteq \alpha - \phi'$, and $\alpha - \phi' \triangleright \phi$ by Definition 27.
- [E3] Let $\alpha \triangleright \phi$ and $\phi \hookrightarrow \phi'$. It is enough to consider the case when $\exists \phi'': \phi \hookrightarrow \phi''$
 11 $\hookrightarrow \phi'$. By Lemma 34 and the Affine Extraction Theorem, there is a canonical
 $P'v' \in \phi'$ such that $P' = P + v$, v creates v' , and $Pv \in \phi$. Further, by Definition 27,
 13 $\exists Q: FAM(Q) \subseteq \alpha$ which discards a canonical element of ϕ . Hence, by Lemmas
 32–30, $ST(Q)$ discards $P'v'$, i.e., $\alpha \triangleright \phi'$.
- [E4] Let $\alpha \cup [\phi] \triangleleft \phi$. By Definition 27, there is a Q such that $FAM(Q) \subseteq \alpha \cup [\phi] \triangleleft$
 15 and Q discards a canonical element Pv of ϕ . By Proposition 11 and Lemma 30,
 Q can be chosen standard. Let $F_2 = FAM(Q) \cap ([\phi] \triangleleft \alpha)$ and $F_1 = FAM(Q) \setminus F_2$.
 17 Since $\alpha \in \mathcal{L}_{\text{fin}}(E)$, $\forall \phi \in F_2, \psi \in F_1: \phi \not\leftrightarrow \psi$. Hence, by Lemma 34, $\exists Q' \in STV(Q)$:
 $Q' = Q_1 + Q_2 \wedge FAM(Q_i) = F_i$. By Lemma 30, Q' discards Pv , and by Lemma 33,
 19 $Q'/P = Q_1/P$, since $[\phi] \triangleleft = FAM(P)$. Hence Q_1 discards Pv and $FAM(Q_1) \subseteq \alpha$,
 i.e., $\alpha \triangleright \phi$.
- 21 It remains to show that $\approx = \approx_{\triangleright}$. Indeed, by Definition 27, $\alpha \approx \beta$ iff $\alpha \cap \beta \triangleright \alpha \cup \beta \setminus \alpha$
 $\cap \beta$, and by Proposition 21, the latter holds iff $\alpha \approx_{\triangleright} \beta$. \square

23 4.2. Translation of DEESs into ASDRSs and AZDFSs

25 The following translation of DEESs into ASDRSs uses a techniques resembling the
 representation of events in a PES as prime intervals of its configuration domain, used
 for construction of PESs from stable families of configurations in [37].

27 **Definition 36.** With a DEES $\mathcal{C} = (E, \leq, \triangleright, \approx)$ we associate an ASDRS $\mathcal{R}_{\mathcal{C}}$ as follows:

- 29 • The terms of $\mathcal{R}_{\mathcal{C}}$ are Lévy-equivalence classes $\langle \alpha \rangle_L, \langle \beta \rangle_L, \dots$ of configurations of \mathcal{C} .
- 31 • The reduction relation of $\mathcal{R}_{\mathcal{C}}$ consists of sets of pairs $u = (\langle \alpha \rangle_L e)$, where $\alpha, \alpha + e \in$
 $\mathcal{L}_{\text{fin}}(E)$ and $\alpha \not\approx \alpha + e$. Pairs $u = (\langle \alpha \rangle_L, e)$ such that $\alpha \approx \alpha + e$ are identified with the
 empty redex $\emptyset = (\langle \alpha \rangle_L, \emptyset)$ in $\langle \alpha \rangle_L$. A non-empty redex $(\langle \alpha \rangle_L, e)$ will also be called
 an *e-redex*.
- 33 • The residual relation is defined as follows: if $u = (\langle \alpha \rangle_L, e)$ and $v = (\langle \alpha \rangle_L, e')$, then
 $u/v = (\langle \alpha + e' \rangle_L, e)$. (Thus $u/v = \emptyset$ iff $\alpha + e' \approx \alpha + e' + e$.)

35 Next we show that the definition above is correct.

Theorem 37. Let $\mathcal{C} = (E, \leq, \triangleright, \approx)$ be a DEES. Then $\mathcal{R}_{\mathcal{C}}$ is an ASDRS.

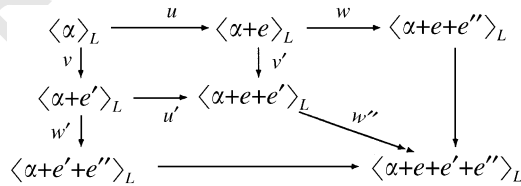
37 **Proof.** For any step $v: \langle \alpha \rangle_L \xrightarrow{(\langle \alpha \rangle_L, e)} \langle \alpha + e \rangle_L$ in $R_{\mathcal{C}}$, and any redex $u = (\langle \alpha \rangle_L, e')$ in
 $\langle \alpha \rangle_L$, u/v is either a (non-empty) redex in $\langle \alpha + e \rangle_L$, or $u/v = \emptyset$. Also, for any other

1 redex $w = (\langle \alpha \rangle_L, e'')$ in $\langle \alpha \rangle_L$ such that $w/v \neq \emptyset$, $w/v = (\langle \alpha + e \rangle_L, e'') \neq (\langle \alpha + e \rangle_L, e') = u/v$,
 i.e., no redex in $\langle \alpha + e \rangle_L$ is a residual of more than one redex in $\langle \alpha \rangle_L$. Further,
 3 $v/v = (\langle \alpha + e \rangle_L, e) = \emptyset$. So to show that $R_{\mathcal{C}}$ is an ASDRS, it remains to show [FD],
 [weak acyclicity], and [stability].

5 [FD] Let U be a set of redexes in $\langle \alpha \rangle_L$, let e_0, \dots, e_{l-1} be an enumeration of U ,
 and let $\alpha_n = \alpha + e_0 + \dots + e_{n-1}$ ($0 \leq n \leq l$). Then a complete develop-
 7 ment of U has a form $P: \langle \alpha_0 \rangle_L \xrightarrow{(\langle \alpha_0 \rangle_L, e_0)} \langle \alpha_1 \rangle_L \xrightarrow{(\langle \alpha_1 \rangle_L, e_1)} \dots \rightarrow \langle \alpha_k \rangle_L$, where for any
 $m = k, \dots, l-1$, $(\langle \alpha_0 \rangle_L, e_m)/P = \emptyset$, that is, $\langle \alpha_k \rangle_L = \langle \alpha_k + e_m \rangle_L$. We show by induc-
 9 tion on $j = l - k$ that $\alpha_k \approx \alpha_k + e_k + \dots + e_{l-1}$. The case $j = 1$ (i.e., $\alpha_k \approx \alpha_k + e_k$)
 is immediate. By the induction assumption, $\alpha_k \approx \alpha_{l-2} \approx \alpha_{l-2} + e_{l-2} \approx \alpha_{l-2} + e_{l-1}$.
 11 By [P2], $\alpha_{l-2} + e_{l-2} + e_{l-1} \approx \alpha_{l-2} + e_{l-1}$. Hence, $\alpha_k \approx \alpha_{l-2} + e_{l-2} + e_{l-1} = \alpha_l$.
 Thus every complete development of U ends at $\langle \alpha_k \rangle_L = \langle \alpha_l \rangle_L$. Further, for any
 13 $u = (\langle \alpha \rangle_L, e)$, $u/P = (\langle \alpha_k \rangle_L, e)$, i.e., u/U does not depend on the particular complete
 development of U , hence [FD] holds.

15 [weak acyclicity] Let $u = (\langle \alpha \rangle_L, e)$, $v = (\langle \alpha \rangle_L, e')$, $e \neq e'$, and suppose $u/v = v/u$
 $= \emptyset$. That is, $\alpha + e + e' \approx \alpha + e \approx \alpha + e'$. Then, by [P1], $\alpha + e \approx \alpha + e' \approx \alpha$, i.e.,
 17 $u = v = \emptyset$.

[stability] Suppose on the contrary that $R_{\mathcal{C}}$ is not stable. Then there are $u \neq v$,
 19 w , w' and w'' such that w and w' are created by u and v respectively, and
 $w'' = w/v' = w'/u' \neq \emptyset$, where $u' = u/v$ and $v' = v/u$. That is, there are $u = (\langle \alpha \rangle_L, e)$,
 21 $v = (\langle \alpha \rangle_L, e')$, $w = (\langle \alpha + e \rangle_L, e'')$, $w' = (\langle \alpha + e' \rangle_L, e'')$, and $w'' = (\langle \alpha + e + e' \rangle_L, e'')$
 such that $\langle \alpha \rangle_L, \langle \alpha + e \rangle_L$ and $\langle \alpha + e + e' \rangle_L$ are different; so are $\langle \alpha \rangle_L, \langle \alpha + e' \rangle_L$ and
 23 $\langle \alpha + e' + e'' \rangle_L$; and $\langle \alpha + e + e' \rangle_L \neq \langle \alpha + e + e' + e'' \rangle_L$ (see the diagram). Further,
 $(\langle \alpha \rangle_L, e'')$ is not a redex since w is a created redex. Hence either $\alpha + e'' \notin \mathcal{L}_{\text{fin}}(\mathcal{C})$,
 25 or $\alpha + e'' \in \mathcal{L}_{\text{fin}}(\mathcal{C})$ and $\alpha \approx \alpha + e''$; but the latter would imply by [P2] that $\alpha +$
 $e + e'' \approx \alpha + e$, contradicting the fact that w is a non-empty redex. Thus $\alpha + e + e''$
 27 and $\alpha + e' + e''$ are both configurations, but their intersection $\alpha + e''$ is not—a
 contradiction, since the intersection of left-closed subsets remains left-closed.



29

The theorem is now proved. \square

31 Clearly, an initial reduction in an ASDRS $\mathcal{R}_{\mathcal{C}}$ can be represented as a left-closed
 enumeration e_1, \dots, e_k of a configuration α where no event e_i in the sequence is *vacuous*,
 33 i.e., $\forall i: \{e_1, \dots, e_{i-1}\} \not\vdash e_i$; the event e_i represents the *non-empty* redex $(\{e_1, \dots, e_{i-1}\},$
 35 $e_i)$. We call such a sequence also an *initial reduction* or an α -*reduction* in the DEES
 \mathcal{C} , written $[\alpha]$. Thus we have the following lemma:

1 **Lemma 38.** *Let \mathcal{C} be a DEES, and let $\mathcal{R}_{\mathcal{C}}$ be its corresponding ASDRS. Then there is a 1–1 correspondence between initial reductions in \mathcal{C} and $\mathcal{R}_{\mathcal{C}}$.*

3 Note that, unlike PESSs, not every configuration in a DEES has a reduction. For
 5 example, consider the DEES consisting of four events $\{e_1, e_2, e'_1, e'_2\}$, ordered as fol-
 7 lows: $e_1 < e'_1$ and $e_2 < e'_2$, and let \triangleright be given by: $\alpha \triangleright e'_1$ iff $e_2 \in \alpha$, and $\alpha \triangleright e'_2$ iff
 9 $e_1 \in \alpha$. Then $\{e_1, e_2, e'_1, e'_2\}$ is a configuration, but any of its left-closed enumera-
 tions, such as e_1, e_2, e'_1, e'_2 , ends either with e'_1 or with e'_2 , and we have $\{e_1, e_2, e'_2\} \triangleright e'_1$
 and $\{e_1, e_2, e'_1\} \triangleright e'_2$. This is not surprising as the vacuous events in left-closed se-
 quences correspond to empty reductions in the corresponding ASDRS, performed in
 the ‘garbage’ (erased redexes). However, we have the following lemma:

11 **Lemma 39.** *Let $\alpha \in \mathcal{L}_{\text{fin}}(E)$, in a DEES. Then any left-closed enumeration of $ST(\alpha)$
 is a reduction.*

13 **Proof.** Let e_0, \dots, e_n be a left-closed enumeration of $ST(\alpha)$. If there was an i such that
 $e_0, \dots, e_{i-1} \triangleright e_i$, then $\alpha \triangleright e_i$ by [E1], contradicting $e_i \in ST(\alpha)$. \square

15 **Definition 40.** With a DEES $\mathcal{C} = (E, \leq, \triangleright, \approx)$, we associate a DFS $\mathcal{F}_{\mathcal{C}} = (\mathcal{R}_{\mathcal{C}}, \simeq_{\mathcal{C}}, \hookrightarrow_{\mathcal{C}})$
 as follows:

- 17 • $\mathcal{R}_{\mathcal{C}}$ is the DRSs defined in Definition 36.
 19 • $\forall e \in E$, we define the $\simeq_{\mathcal{C}}$ -family ϕ_e corresponding to e , or the e -family, as the set
 of all e -redexes $([\alpha], e)$. That is, $([\alpha], e) \simeq_{\mathcal{C}} ([\beta], e')$ iff $e = e'$.
 • Define $\hookrightarrow_{\mathcal{C}}$ by: $\phi_e \hookrightarrow_{\mathcal{C}} \phi_{e'}$ iff $e < e'$.

21 **Theorem 41.** *Let $\mathcal{C} = (E, \leq, \triangleright, \approx)$ be a DEES, let $\mathcal{F}_{\mathcal{C}} = (\mathcal{R}_{\mathcal{C}}, \simeq_{\mathcal{C}}, \hookrightarrow_{\mathcal{C}})$, and let $\mathcal{F}_{\mathcal{R}_{\mathcal{C}}} = (\mathcal{R}_{\mathcal{C}}, \simeq_z, \hookrightarrow_z)$. Then $\mathcal{F}_{\mathcal{C}} = \mathcal{F}_{\mathcal{R}_{\mathcal{C}}}$, hence $\mathcal{F}_{\mathcal{C}}$ is an AZDFS.*

23 **Proof.** We need to show that (a) $\simeq_z = \simeq_{\mathcal{C}}$ and (b) $\hookrightarrow_z = \hookrightarrow_{\mathcal{C}}$.

(a) By Definition 36, $([\alpha], e) \simeq_z ([\beta], e') \Rightarrow e = e'$, implying $\simeq_z \subseteq \simeq_{\mathcal{C}}$. For the con-
 25 verse, consider two redexes $([\alpha], e) \simeq_{\mathcal{C}} ([\beta], e)$. Then $([\alpha \cap \beta], e)$ is also a redex
 since $\alpha \cap \beta + e = \alpha + e \cap \beta + e \in \mathcal{L}_{\text{fin}}(E)$, and $([\alpha], e)$ and $([\beta], e)$ are its residuals.
 27 (Note that $(\langle \alpha \cap \beta \rangle_L, e) \neq \emptyset$ since $(\langle \alpha \cap \beta \rangle_L, e) = \emptyset \Rightarrow (\langle \alpha \rangle_L, e) = \emptyset$ by [P2].) Hence
 $([\alpha], e) \simeq_z ([\beta], e)$.

(b) In order to show that $\phi_e \hookrightarrow_{\mathcal{C}} \phi_{e'} \Leftrightarrow \phi_e \hookrightarrow_z \phi_{e'}$, we need to show that $e < e'$ iff
 29 $e \in [\gamma]$ for any $([\gamma], e') \in \phi_{e'}$ (by (a), $\phi_e, \phi_{e'}$ are families both w.r.t. $\simeq_{\mathcal{C}}$ and \simeq_z).
 31 Sufficiency is immediate since $\gamma + e'$ is left closed. For the converse, we have from
 Theorem 13 that $([\lceil e' \rceil], e')$ is a canonical element of $\phi_{e'}$ (when $\phi_{e'}$ is considered
 33 as a \simeq_z -family), hence $e < e'$. \square

4.3. Equivalence of the transition and event models

35 We now establish an equivalence between DEESs and AZDFSs, implying together
 with the results of previous sections equivalence of the considered four transition and
 37 event models—ASDRSs, AZDFSs, DEESs and DPESs.

1 Let \mathcal{R} be a DRS. Let us define the *reduction tree* of \mathcal{R} to be a tree whose nodes
 2 are labelled with terms in \mathcal{R} , whose root is labelled with t_0 , the initial term of \mathcal{R} ,
 3 and whose arcs correspond to redexes in \mathcal{R} . Thus, sons of a node are target terms of
 4 redexes whose source term is the label of that node. (Reduction trees are considered
 5 up to the order of arcs coming out of a node: the order in which the arcs are drawn
 6 is irrelevant.) Note that different nodes can be labelled with the same term, so even
 7 if \mathcal{R} has cycles, its reduction tree is a tree. Thus, the branches of the reduction tree
 8 of \mathcal{R} are in 1–1 correspondence with initial reductions in \mathcal{R} . We call two such trees
 9 *isomorphic* if the trees obtained by removing all labels are identical. Hence, we can
 10 define two DRSs to be *isomorphic* if there is an isomorphism between their reduction
 11 trees preserving the residual relation. An isomorphism between two DFSs must further
 12 preserve the family and contribution relations.

13 **Lemma 42.** *Let \mathcal{F} be an AZDFS and let $\mathcal{C}_{\mathcal{F}}$ be its corresponding DEES.*

- 14 (1) *For any initial reduction $[\alpha] = e_0, e_1, \dots, e_n$ in $\mathcal{C}_{\mathcal{F}}$ there is a unique initial re-*
 15 *duction $g([\alpha]) = P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \dots \xrightarrow{u_{n-1}} t_n$ in \mathcal{F} such that $Fam(P_i u_i) = e_i$, where*
 16 *$P_i : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \dots \xrightarrow{u_{i-1}} t_i$. (Thus $FAM(P) = \alpha$.)*
- 17 (2) *Let $P : t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \dots \xrightarrow{u_{n-1}} t_n$ in \mathcal{F} , and let $e_i = \phi_i = Fam(P_i u_i)$, where $P_i : t_0$
 18 $= t_0 \xrightarrow{u_0} t_1 \xrightarrow{u_1} \dots \xrightarrow{u_{i-1}} t_i$. Then $f(P) = e_0, e_1, \dots, e_n$ is a reduction in $\mathcal{C}_{\mathcal{F}}$.*
- 19 (3) *The translation functions g and f between reductions in \mathcal{F} and $\mathcal{C}_{\mathcal{F}}$ commute:*
 20 *for any initial reduction P in \mathcal{F} , $g(f(P)) = P$, and for any reduction $[\alpha]$ in $\mathcal{C}_{\mathcal{F}}$,*
 21 *$f(g([\alpha])) = [\alpha]$.*

Proof. (1) By definition of a reduction in a DEES, e_0, e_1, \dots, e_n is left-closed, and
 22 $\{e_0, \dots, e_{j-1}\} \not\triangleright e_j$ for any $j = 0, \dots, n$. Take for u_0 the redex in t_0 such that $Fam(\emptyset u_0)$
 23 $= e_0$; existence of such a redex is immediate from the definition of $\mathcal{C}_{\mathcal{F}}$, and uniqueness
 24 follows from [initial]. Suppose u_1, \dots, u_{i-1} are already defined so that $Fam(P_k u_k) = e_k$
 25 for all $1 \leq k < i$. We need to show that $\exists u_i \in t_i$ such that $Fam(P_i u_i) = e_i$; uniqueness of
 26 such a redex follows from the separability of \mathcal{F} (see Theorem 12) and from the fact
 27 that in separable AZDFSs, a term may contain at most one redex of a given family (see
 28 Lemma 6.1 in [18], or Lemma 5.2 in [20]). Let $N_i v_i$ be a canonical element of e_i which
 29 exists by the Affine Extraction Theorem. By the same theorem, N_i contracts exactly one
 30 redex in every contributor family of e_i (and does not contract any other redex). Thus,
 31 by [contribution], $FAM(N_i) \subseteq FAM(P_i)$, and $N_i \leq P_i$ by Proposition 11. Now we can
 32 take $u_i = v_i / (P_i / N_i)$. If on the contrary $u_i = \emptyset$, then P_i would discard $N_i v_i$ (since P_i / N_i
 33 is external to v_i as $Fam(N_i v_i) \notin FAM(P_i)$), and we would have by Definition 27 that
 34 $\{e_0, \dots, e_{i-1}\} \triangleright e_i$, contradicting the assumption. Thus a reduction P as above exists,
 35 and is unique.

36 (2) We need to show that $f(P) = e_0, e_1, \dots, e_n$ is left-closed and does not contain
 37 vacuous events (by Definition 27, $\phi_i = e_i$ are events in $\mathcal{C}_{\mathcal{F}}$). By [contribution], P_i
 38 contracts at least one redex in every contributor family of ϕ_i , hence $f(P)$ is left-closed.
 39 If on the contrary say ϕ_j is vacuous, i.e., $\{\phi_0, \dots, \phi_{j-1}\} \triangleright \phi_j$, then by Definition 27
 40 there is an initial reduction Q such that $FAM(Q) \subseteq \{\phi_0, \dots, \phi_{j-1}\}$ and Q discards a

- 1 canonical element $N_j v_j$ of ϕ_j . By Proposition 11, $N_j, Q \trianglelefteq P_j$, and by Definition 6,
 P_j erases $N_j v_j$, i.e., $v_j / (P_j / N_j) = \emptyset$. Hence $u_j / ((N_j + v_j) / P_j) = u_j / \emptyset = u_j$, contradicting
 3 Lemma 33 (since $N_j v_j \simeq_z P_j u_j$).

(3) Immediate from the constructions in (1) and (2). \square

- 5 **Theorem 43** (Equivalence of AZDFSs and DEESs). (1) For any DEES \mathcal{C}, \mathcal{C} and $\mathcal{C}_{\mathcal{F}_e}$
 are isomorphic.

- 7 (2) For any AZDFS \mathcal{F}, \mathcal{F} and $\mathcal{F}_{\mathcal{C}_{\mathcal{F}}}$ are isomorphic.

Proof. (1) Let $\mathcal{C} = (E, \leq, \triangleright, \approx)$ and $\mathcal{C}' = \mathcal{C}_{\mathcal{F}_e} = (E', \leq', \triangleright', \approx')$. By Definitions 40
 9 and 27, $E' = \{\phi_e \mid e \in E\}$, where ϕ_e is the e -family of \mathcal{F}_e . We show that $f : e \mapsto \phi_e$ is
 an isomorphism between \mathcal{C} and \mathcal{C}' . Since f clearly an isomorphism between E and
 11 E' , and since \mathcal{C}' is a DEES by Theorems 35 and 41, we need to show that (a) $e \leq e'$
 iff $\phi_e \leq' \phi_{e'}$; and (b) $\alpha \approx \beta$ iff $\phi_\alpha \approx' \phi_\beta$, where $\phi_\alpha = \{\phi_e \mid e \in \alpha\}$ and $\phi_\beta = \{\phi_e \mid e \in \beta\}$.

13 (a) Immediate from Definitions 40 and 27.

(b) By Definition 27, $\phi_\alpha \approx' \phi_\beta$ iff $\forall e \in \alpha \cup \beta \setminus \alpha \cap \beta$, there is a reduction $[\gamma]$ in \mathcal{F}_e such
 15 that $FAM([\gamma]) \subseteq \phi_\alpha \cap \phi_\beta$ and $[\gamma]$ discards a canonical element $[[e]^{<} e]$ of ϕ_e (see
 Theorem 13). By Definitions 6 and 36, the latter means that $e \notin \gamma$, $\gamma \subseteq \alpha \cap \beta$, and
 17 $\gamma \cup [e]^{<} \approx \gamma \cup [e]^{<}$, thus by Definition 16, and Lemmas 39 and 19 we need to
 prove that $\alpha \approx \beta$ iff $\exists \text{standard } \gamma : \gamma \subseteq \alpha \cap \beta \wedge \gamma \triangleright \alpha \cup \beta \setminus \alpha \cap \beta$. But this is immediate
 19 from Proposition 21 (since $\approx = \approx_{\triangleright}$) and Lemma 19.

(2) By Theorems 41 and 35, $\mathcal{F}_{\mathcal{C}_{\mathcal{F}}}$ is an AZDFS, hence it is enough to show that
 21 the underlying ASDRSs \mathcal{R} and $\mathcal{R}_{\mathcal{C}_{\mathcal{F}}}$ of \mathcal{F} and $\mathcal{F}_{\mathcal{C}_{\mathcal{F}}}$ are isomorphic. This follows
 immediately from Lemmas 42 and 38. \square

23 5. Lévy's configuration domains

Next, we define Lévy's configuration domains in DEESs and show that they are
 25 isomorphic to Lévy's reduction spaces in corresponding ASDRSs.

Definition 44. (1) Let \mathcal{C} be a DEES. Lévy's configuration domain $\mathcal{L}_{\text{fin}}^{\trianglelefteq}(\mathcal{C}) = (\mathcal{L}_{\text{fin}}^{\approx}(\mathcal{C}),$
 27 $\trianglelefteq)$, where $\mathcal{L}_{\text{fin}}^{\approx}(\mathcal{C}) = \mathcal{L}_{\text{fin}}(\mathcal{C}) / \approx = \{\langle \alpha \rangle_L \mid \alpha \in \mathcal{L}_{\text{fin}}(\mathcal{C})\}$ and \trianglelefteq is a partial order defined
 by $\langle \alpha \rangle_L \trianglelefteq \langle \beta \rangle_L$ iff $\exists \beta' \in \langle \beta \rangle_L : \alpha \subseteq \beta'$.

(2) Let \mathcal{R} be an ASDRS, and let $\mathcal{L}_{\text{fin}}(\mathcal{R})$ be the set of initial reductions in \mathcal{R} . Lévy's
 29 reduction space $\mathcal{L}_{\text{fin}}^{\trianglelefteq}(\mathcal{R}) = (\mathcal{L}_{\text{fin}}^{\approx}(\mathcal{R}), \trianglelefteq)$, where $\mathcal{L}_{\text{fin}}^{\approx}(\mathcal{R}) = \mathcal{L}_{\text{fin}}(\mathcal{R}) / \approx = \{\langle P \rangle_L \mid P \in$
 31 $\mathcal{L}_{\text{fin}}(\mathcal{R})\}$, and $\langle P \rangle_L \trianglelefteq \langle Q \rangle_L$ iff $P \trianglelefteq Q$.

Note that if $\langle \alpha \rangle_L \trianglelefteq \langle \beta \rangle_L$, $\alpha \approx \alpha'$ and $\beta \approx \beta'$, then $\langle \alpha' \rangle_L \trianglelefteq \langle \beta' \rangle_L$, implying that
 33 Definition 44(1) is correct: Indeed, $\langle \alpha \rangle_L \trianglelefteq \langle \beta \rangle_L$ iff $\exists \beta'' \approx \beta : \alpha \subseteq \beta''$, and by Proposi-
 tion 21 and [E1], $\alpha \approx \alpha' \Rightarrow \alpha \triangleright (\alpha' \setminus \alpha) \Rightarrow \beta'' \triangleright (\alpha' \setminus \alpha) \Rightarrow \beta' \approx \beta \approx \beta'' \approx \beta'' \cup (\alpha' \setminus \alpha) =$
 $\beta'' \cup \alpha'$, thus $\langle \alpha' \rangle_L \trianglelefteq \langle \beta' \rangle_L$.

1 **Lemma 45.** *Let P be an initial reduction in an AZDFS \mathcal{F} . Then, in the corresponding DEES $\mathcal{C}_{\mathcal{F}}$, $FAM(ST(P)) = ST(FAM(P))$.*

3 **Proof.** First note that, by [contribution] and the definition of \leq (Definition 27), for any initial reduction Q , $FAM(Q)$ is indeed a configuration.

5 (\subseteq) Let $\phi \in FAM(ST(P))$. By Proposition 11, $\phi \in SFAM(P)$, i.e., there is $P'v' \in \phi$ that is contracted in P and is P -needed, thus is P -essential. Let P^*v^* be a canonical element of ϕ . Then it is P -essential by Lemma 32. For any initial reduction Q with $FAM(Q) \subseteq FAM(P)$, $Q \triangleleft P$ by Proposition 11, thus P^*v^* is Q -essential too. Hence Q does not discard P^*v^* , implying by Definition 27 that $FAM(P) \not\triangleright \phi$, i.e., $\phi \in ST(FAM(P))$.

11 (\supseteq) Let $\phi \in ST(FAM(P))$. Then $\phi \in FAM(P)$ and $FAM(P) \not\triangleright \phi$, which by Definition 27 means that there is no initial Q with $FAM(Q) \subseteq FAM(P)$ such that Q discards a canonical element P^*v^* of ϕ . In particular, $ST(P)$ does not discard P^*v^* , and by Lemmas 32 and 7(1), the redex $P'v' \in \phi$ contracted in P is P -needed, i.e., $\phi \in SFAM(P)$; thus $\phi \in FAM(ST(P))$ by Proposition 11. \square

17 **Lemma 46.** *Let Q, Q' be initial reductions in an AZDFS \mathcal{F} . Then $Q \approx Q'$ in \mathcal{F} iff $FAM(Q) \approx FAM(Q')$ in the corresponding DEES $\mathcal{C}_{\mathcal{F}} = (E_{\mathcal{F}}, \leq, \triangleright, \approx)$.*

Proof. (\Rightarrow) By Definition 27, we need to show that $\forall \phi \in FAM(Q) \cup FAM(Q') \setminus FAM(Q) \cap FAM(Q')$, an initial reduction P with $FAM(P) \subseteq FAM(Q) \cap FAM(Q')$ discards a canonical element Nv of ϕ . Take $P = ST(Q)$. By Proposition 11, $FAM(P) \subseteq FAM(Q) \cap FAM(Q')$. Let Q^*v^* be an element of ϕ contracted in Q . Since by Proposition 11 $\phi \notin FAM(P) = SFAM(Q)$, Q^*v^* is Q -unneded, hence Q -inessential by Lemma 7 (1). Hence, by Lemma 30, P discards Q^*v^* . By the Affine Extraction Theorem, $\exists N': N + N' \approx Q^*$ and $v^* = v/N'$. Since $Q^* \leq Q \approx P$, $N'/(P/N) = \emptyset$, hence $v/(P/N) = v^*/(P/Q^*) = \emptyset$ by the Cube Lemma. Since $\phi \notin FAM(P)$, P/N is external to v , thus P discards Nv .

27 (\Leftarrow) By Proposition 25, $FAM(Q) \approx FAM(Q')$ implies that $ST(FAM(Q)) = ST(FAM(Q'))$; hence, by Lemma 45, $FAM(ST(Q)) = FAM(ST(Q'))$, implying by Proposition 11 that $Q \approx Q'$. \square

31 **Lemma 47.** *Let \mathcal{F} be an AZDFS, let e_0, e_1, \dots, e_n be a left-closed enumeration of $\alpha \in \mathcal{L}_{\text{fin}}(\mathcal{C}_{\mathcal{F}})$, and let e_{i_0}, \dots, e_{i_m} be obtained from e_0, e_1, \dots, e_n by eliminating all vacuous events, i.e., $e_j \in \{e_{i_0}, \dots, e_{i_m}\}$ iff $\{e_0, \dots, e_{j-1}\} \not\triangleright e_j$. Then e_{i_0}, \dots, e_{i_m} is a reduction in $\mathcal{C}_{\mathcal{F}}$ and $\{e_{i_0}, \dots, e_{i_m}\} \approx \{e_0, \dots, e_n\}$.*

Proof. If there were l, k such that $e_l < e_{i_k}$ (hence $l < i_k$) and $l \neq i_0, \dots, i_{k-1}$, then $\{e_0, \dots, e_{l-1}\} \triangleright e_l$, $\{e_0, \dots, e_{i_k-1}\} \not\triangleright e_{i_k}$, and by [E1], $\{e_0, \dots, e_{i_k-1}\} \triangleright e_l$, contradicting [E3]. Hence e_{i_0}, \dots, e_{i_m} is left-closed. Since $\{e_0, \dots, e_{i_k-1}\} \not\triangleright e_{i_k}$, $\{e_{i_0}, \dots, e_{i_{k-1}}\} \not\triangleright e_{i_k}$ by [E1], thus e_{i_0}, \dots, e_{i_m} is a reduction in $\mathcal{C}_{\mathcal{F}}$. Finally, $\{e_{i_0}, \dots, e_{i_m}\} \approx \{e_0, \dots, e_n\}$ follows from Proposition 21. \square

1 **Proposition 48.** *Let P, Q be initial reductions in an ASDRS \mathcal{R} . Then $\langle P \rangle_L \sqsubseteq \langle Q \rangle_L$ iff $\langle FAM(P) \rangle_L \sqsubseteq \langle FAM(Q) \rangle_L$ in $\mathcal{C}_{\mathcal{R}}$.*

3 **Proof.** (\Rightarrow) By Proposition 11, $\langle P \rangle_L \sqsubseteq \langle Q \rangle_L$ iff $\exists Q' \approx Q: FAM(P) \subseteq FAM(Q')$, which
 by Definition 44 holds iff $\langle FAM(P) \rangle_L \sqsubseteq \langle FAM(Q') \rangle_L$ in $\mathcal{C}_{\mathcal{R}}$. But $Q \approx Q'$ implies
 5 $FAM(P) \approx FAM(Q')$ in $\mathcal{C}_{\mathcal{R}}$ by Lemma 46, and therefore $\langle FAM(P) \rangle_L \sqsubseteq \langle FAM(Q) \rangle_L$
 in $\mathcal{C}_{\mathcal{R}}$.

7 (\Leftarrow) By Definition 44, $\exists \alpha: \alpha \approx FAM(Q) \wedge FAM(P) \subseteq \alpha$; thus by Lemma 47 $\exists [\beta]:$
 $FAM(P) \subseteq \beta \subseteq \alpha \wedge \beta \approx \alpha$ (assuming that say e_0, e_1, \dots, e_n is an enumeration of α whose
 9 initial part is an enumeration of $FAM(P)$, we can take e_{i_0}, \dots, e_{i_m} as in Lemma 47
 for $[\beta]$). By Lemma 42, there is a Q' such that $FAM(Q') = \beta$ and P is an initial part
 11 of Q' . By Lemma 47, $Q \approx Q'$, hence $P \sqsubseteq Q$, therefore $\langle P \rangle_L \sqsubseteq \langle Q \rangle_L$. \square

Theorem 49. *Let \mathcal{R} be an ASDRS and let $\mathcal{C} = \mathcal{C}_{\mathcal{R}}$ be its corresponding DEES. Then
 13 $f: \langle P \rangle_L \mapsto \langle FAM(P) \rangle_L$ is the isomorphism between $\mathcal{L}_{\text{fin}}^{\sqsubseteq}(\mathcal{R})$ and $\mathcal{L}_{\text{fin}}^{\sqsubseteq}(\mathcal{C})$, with the
 inverse $f^{-1}: \langle \alpha \rangle_L \mapsto \langle [ST(\alpha)] \rangle_L$. Moreover, reductions in $\langle P \rangle_L$ and $\langle FAM(P) \rangle_L$ are in
 15 one-to-one correspondence.*

Proof. By Lemmas 46 and 39 and Proposition 25, f and f^{-1} are well defined. The
 17 rest is immediate from Lemma 42 and Proposition 48. \square

A similar statement is valid when \mathcal{C} is an arbitrary DEES and $\mathcal{R} = \mathcal{R}_{\mathcal{C}}$: We only
 19 need to replace $FAM(P)$ in the definition of f with $EVN(P)$, where $EVN(P)$ is the
 set of events fired in P .

21 6. Preservation of neededness and stability by the translations

This section develops a normalization-by-neededness theory for DEESs, and relates it
 23 with that in affine SDRSs, establishing a relationship between the concepts of stability
 intrinsic to the above models, and between neededness and stability in each model.

In order to extend the theory of Relative Normalization [9–11] from SDRSs to
 25 DEESs, we can restrict ourselves to ASDRSs of the form $\mathcal{R}_{\mathcal{C}}$, where \mathcal{C} is a DEES: by
 27 Theorem 49, such ASDRSs contain anisomorphic copy of any ASDRS.

Definition 50 (Glauert et al. [9]). Let \mathcal{S} be a set of terms in a DRS \mathcal{R} .

29 (1) We call a redex $u \subseteq t$ \mathcal{S} -needed if $t \notin \mathcal{S}$ and at least one residual of u is contracted
 in any reduction from t to a term in \mathcal{S} ; we and call u \mathcal{S} -unneeded otherwise.

31 (2) We call \mathcal{S} stable⁵ if:

- (a) \mathcal{S} is closed under reduction; and
- 33 (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.

⁵ Our definition follows [9] rather than [10, 11]; in the latter papers, the definition of stability of sets allows a slightly weaker condition than closure under reduction.

1 All interesting sets of ‘finite results’, such as of normal forms, head-normal forms,
 3 weak-head-normal forms, etc., are stable [10]. It has been shown in [11] (a corollary
 of the so-called Relative Normalization theorem) that, for any stable set \mathcal{S} of terms in
 an ASDRS \mathcal{R} and any \mathcal{S} -normalizable term $t \notin \mathcal{S}$, t contains at least one \mathcal{S} -needed
 5 redex, and repeated contraction of \mathcal{S} -needed redexes is \mathcal{S} -normalizing, which justifies
 the concept of stable sets.

7 Our translation of AZDFSs into DEESs suggests the following definition of relative
 neededness and stability in DEESs.

9 **Definition 51.** Let \mathcal{S}' be a set of configurations in a DEES $\mathcal{C} = (E, \leq, \triangleright, \approx)$.

- (1) We call $e \in E$ \mathcal{S}' -needed, written $NE'_{\mathcal{S}'}(e)$ if, for any $\alpha \in \mathcal{S}'$, $e \in \alpha$, and call it
 11 \mathcal{S}' -unneeded otherwise.
 (2) We call \mathcal{S}' stable if:
 13 (a) \mathcal{S}' is closed under inclusion.
 (b) \mathcal{S}' is closed under unneeded expansion:
 15 $\alpha + e \in \mathcal{S}' \wedge \alpha \notin \mathcal{S}' \Rightarrow NE'_{\mathcal{S}'}(e)$; and
 (c) \mathcal{S}' is closed under permutation equivalence.

17 Note that the relative neededness concept already makes sense for PESs.

19 The next theorem gives two equivalent definitions of stability in DEESs, more in
 the spirit of the definition of stability in PESs [37], and more convenient to generalize
 to the case of infinite configurations.

21 **Theorem 52.** *Equivalent definitions of stability of sets of finite configurations in a*
 23 *DEES can be obtained from Definition 51 by replacing condition (b) with either of*
the following two conditions (b') or (b''):

- (b') \mathcal{S}' is closed under intersection.
 25 (b'') \mathcal{S}' is has a unique \subseteq -minimal element $\alpha'_{\mathcal{S}'}$.

Proof. We show that (i) (b) \Rightarrow (b'); (ii) (b') \Rightarrow (b'') and (iii) (a), (b'') \Rightarrow (b).

27 (i) Let $\alpha, \beta \in \mathcal{S}'$, and suppose on the contrary that $\alpha \cap \beta \notin \mathcal{S}'$. Further, let $e_1, \dots, e_m,$
 29 \dots, e_n be a left-closed enumeration of α such that e_1, \dots, e_m is a left-closed enu-
 meration of $\alpha \cap \beta$ (such an enumeration exists since $\alpha \cap \beta$ is left-closed). Then
 there is $m \leq j < n$ such that $\{e_1, \dots, e_j\} \notin \mathcal{S}'$ and $\{e_1, \dots, e_{j+1}\} \in \mathcal{S}'$. By (b), e_{j+1}
 31 is \mathcal{S}' -needed, which is not possible since $e_{j+1} \notin \beta \in \mathcal{S}'$.

(ii) If \mathcal{S}' does not have a unique \subseteq -minimal element, then it has at least two differ-
 33 ent \subseteq -minimal elements α and β (i.e., $\nexists \gamma \subset \alpha: \gamma \in \mathcal{S}'$, and similarly for β). By
 (b'), $\alpha \cap \beta \in \mathcal{S}'$. But $\alpha \cap \beta \subset \alpha$, contradicting the minimality of α .

35 (iii) Let α, e be such that $\alpha \notin \mathcal{S}' \wedge \alpha + e \in \mathcal{S}'$. By (a) and (b''), this means that
 37 $\alpha'_{\mathcal{S}'} \not\subseteq \alpha \wedge \alpha'_{\mathcal{S}'} \subseteq \alpha + e$. Hence $e \in \alpha'_{\mathcal{S}'}$, and therefore $\forall \delta \in \mathcal{S}': e \in \delta$ (since $\alpha'_{\mathcal{S}'} \subseteq \delta$ by
 (b'')). \square

The proof of the following proposition is routine, and therefore omitted.

1 **Proposition 53.** Let \mathcal{S}' be a set of configurations, in a DEES \mathcal{C} , closed under permutation equivalence, and let \mathcal{S} be the set of corresponding terms in the corresponding
 3 ASDRS $\mathcal{R}_{\mathcal{C}}$ (which are Lévy-equivalence classes of configurations in \mathcal{C}). Then \mathcal{S}' is stable in \mathcal{C} iff \mathcal{S} is stable in $\mathcal{R}_{\mathcal{C}}$. Further, if \mathcal{S}' is stable, an event e is \mathcal{S}' -needed
 5 in \mathcal{C} iff any e -redex is \mathcal{S} -needed in $\mathcal{R}_{\mathcal{C}}$.

We conclude this section with a theorem expressing the relationship between concepts of stability and neededness in DEESs, analogous to the Relative Normalization Theorem for ASDRSs [11]:

9 **Theorem 54.** Let \mathcal{S}' be a stable set of configurations in a DEES \mathcal{C} . Then a repeated execution of \mathcal{S}' -needed events in any configuration $\alpha \notin \mathcal{S}'$, possibly interleaved by
 11 finite sequences of \mathcal{S}' -unneeded steps, eventually yields a configuration in \mathcal{S}' .

13 **Proof.** Immediate from closure of \mathcal{S}' under inclusion (since \mathcal{S}' consists of finite configurations, hence there are only a finite number of \mathcal{S}' -needed events). \square

7. Relativized information domains

15 In this section, we introduce orderings on configurations relativized w.r.t. stable sets of results, in a DEES \mathcal{C} , and show that they form finitary prime algebraic complete
 17 lattices. Unlike the previous sections, we will now allow for infinite configurations—(countably) infinite left-closed sets of events in \mathcal{C} . The set of finite and infinite configurations in \mathcal{C} will be denoted by $\mathcal{L}(\mathcal{C})$.

We start by some preliminary lattice theoretic definitions; we follow [37].

21 **Definition 55.**

- A *complete lattice* is a partial order $\mathcal{L} = (L, \sqsubseteq)$ which has least upper bounds (joins or suprema) $\sqcup X$ and greatest lower bounds (meets or infima) $\sqcap X$ of arbitrary subsets $X \subseteq L$. We write $x \sqcup y$ and $x \sqcap y$ for the join and meet, respectively, of two elements $x, y \in L$.
- A *directed subset* of a partial order \mathcal{L} is a subset $Y \subseteq L$ with the property that for any finite set $X \subseteq Y$ there is an element $y \in Y$ such that $\forall x \in X. x \sqsubseteq y$.
- A *finite element* of a complete lattice is an element z with the property that, for all directed subsets Y , if $z \sqsubseteq \sqcup Y$, then there is some $y \in Y$ for which $z \sqsubseteq y$.
- A complete lattice \mathcal{L} is *algebraic* if for any element $d \in L$ the set $\{x \sqsubseteq d \mid x \text{ is finite}\}$ is directed and has least upper bound d .
- A *complete prime* of a complete lattice \mathcal{L} is an element $p \in L$ such that $p \sqsubseteq \sqcup X \Rightarrow \exists x \in X. p \sqsubseteq x$ for any subset $X \subseteq L$. \mathcal{L} is *prime algebraic* if, for all $x \in L$, $x = \sqcup \{p \sqsubseteq x \mid p \text{ is a complete prime}\}$.
- An algebraic lattice is *finitary* iff every finite element dominates only a finite number of elements, i.e., $\{x \mid x \sqsubseteq d\}$ is finite for every finite element d .
- A complete lattice \mathcal{L} is *distributive* if $x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z)$, for all elements $x, y, z \in L$.

1 **Theorem 56** (Winskel [37]). *Let \mathcal{L} be a finitary algebraic lattice. Then \mathcal{L} is prime algebraic iff it is distributive.*

3 We now extend some of the previously introduced concepts from finite to infinite configurations. The concepts of an α -reduction and a \mathcal{S}' -needed event generalize immediately.
5

Definition 57. Let $\mathcal{C} = (E, \triangleleft, \triangleright, \approx)$ be a DEES.

- 7 (1) Let $\alpha \in \mathcal{L}(\mathcal{C})$. Then $\alpha \triangleright e$ iff $\exists \beta \subseteq_{\text{fin}} \alpha : \beta \triangleright e$.
 (2) Let $\alpha, \beta \in \mathcal{L}(\mathcal{C})$. Then $\alpha \approx \beta$ iff $\alpha \triangleright \beta \setminus \alpha \wedge \beta \triangleright \alpha \setminus \beta$.
 9 (3) We call $\mathcal{S}' \subseteq \mathcal{L}(\mathcal{C})$ *stable* if it is closed under \subseteq , has a unique \subseteq -minimal element $\alpha_{\mathcal{S}'}$, and is closed under permutation equivalence.

11 It is easy to check that Definitions 57(1) and (2) do not conflict with the definitions of the corresponding concepts for finite configurations. Further, it is an easy consequence of Definition 57(1) and [E1] that:
13

Lemma 58. *Let $\alpha \triangleright e$ and $\alpha \subseteq \beta \in \mathcal{L}(\mathcal{C})$. Then $\beta \triangleright e$.*

15 We call a reduction in a DEES \mathcal{S}' -needed-fair, or simply \mathcal{S}' -fair, if it fires at least all \mathcal{S}' -needed events. \mathcal{S}' -fair configurations are defined similarly.

17 **Theorem 59.** *Let \mathcal{S}' be a stable set of configurations in a DEES \mathcal{C} . Then its \subseteq -minimal element $\alpha_{\mathcal{S}'}$ consists of all \mathcal{S}' -needed events of \mathcal{C} . Consequently, \mathcal{S}' consists of all \mathcal{S}' -fair configurations, and any \mathcal{S}' -fair reduction generates a configuration in \mathcal{C} .*
19

21 **Proof.** Since $\alpha_{\mathcal{S}'} \in \mathcal{S}'$, all \mathcal{S}' -needed events of \mathcal{C} are in $\alpha_{\mathcal{S}'}$. If on the contrary $\alpha_{\mathcal{S}'}$ contains an \mathcal{S}' -unnneeded event e , then $\exists \beta \in \mathcal{S}' : e \notin \beta$, hence $\beta \not\subseteq \alpha_{\mathcal{S}'}$, contradicting the minimality of $\alpha_{\mathcal{S}'}$. The rest is immediate. \square
23

For example, firing all events at any state, or all \mathcal{S}' -needed events, in parallel yields
25 an \mathcal{S}' -fair reduction since any finite configuration not yet in \mathcal{S}' can fire at least one \mathcal{S}' -needed event.

27 Note that the set of all *fair configurations*—configurations $\alpha \in \mathcal{C}$ such that $\forall e \in \mathcal{C} : e \notin \alpha \Rightarrow \alpha \triangleright e$ —need not be stable for every DEES \mathcal{C} . The reason is that, unlike the finite case, a configuration α need not be equivalent to its standard variant $ST_{\triangleright}(\alpha) = \{e \in \alpha \mid \alpha \not\triangleright e\}$. Indeed, consider the DEES corresponding to the ASDRS from [17] demonstrating the failure of the infinitary standardization theorem: there are infinitely many events $e_0, e'_1, e_1, e'_2, e_2, \dots$, the causal dependency is the transitive reflexive closure of dependencies $e_i < e_{i+1}, e'_{i+1}$ ($i = 0, 1, \dots$), and the erasure relation is defined by $\alpha \triangleright e'_j$ iff $e'_k \in \alpha$ for some $k > j, k = 2, 3, \dots$. Then any fair configuration consists of all non-erasable events e_i , and at least one infinite sequence of erasable events e'_i , and it is immediate
35 that the set of all fair configurations does not have the \subseteq -minimal element. Note that

1 for any fair configuration α in that DEES, $ST(\alpha)$ consists of all non-erasable events e_i ,
 2 and $\alpha \not\approx ST(\alpha)$ since $ST(\alpha) \not\vdash e'_k$ for any k . Thus $ST(\alpha) = ST(\beta)$ does not imply $\alpha \approx \beta$
 3 as $ST(ST(\alpha)) = ST(\alpha)$. However, the converse is always true:

Lemma 60. *Let $\alpha, \beta \in \mathcal{L}(\mathcal{C})$. Then $\alpha \approx \beta$ implies $ST(\alpha) = ST(\beta)$.*

5 **Proof.** By Definition 57(2) $\alpha \triangleright \beta \setminus \alpha \cap \beta$ and $\beta \triangleright \alpha \setminus \alpha \cap \beta$. Hence $ST(\alpha), ST(\beta) \subseteq \alpha \cap \beta$.
 Suppose on the contrary that $\exists e \in ST(\alpha) \setminus ST(\beta)$. Then $\alpha \not\vdash e$ and $\beta \triangleright e$, i.e., $\exists \beta_0 \subseteq_{\text{fin}} \beta$:
 7 $\beta_0 \triangleright e$, implying by Lemma 58 that $\alpha \cap \beta \not\vdash e$ and $\beta' = \beta_0 \cap (\alpha \cap \beta) \not\vdash e$. Therefore, by
 Proposition 21, $\exists e' \in \beta_0 \setminus \beta'$: $\beta' \not\vdash e'$. Thus $\alpha \not\vdash e'$ by Lemma 58 (since $\beta' \subseteq \alpha \cap \beta \subseteq \alpha$)
 9 and $e' \in \beta \setminus \alpha \cap \beta$, contradicting $\alpha \triangleright \beta \setminus \alpha \cap \beta$. \square

Definition 61. Let $\alpha, \beta \in \mathcal{L}(\mathcal{C})$, and let \mathcal{S}' be a stable set of configurations in \mathcal{C} . We
 11 define $\alpha \trianglelefteq_{\mathcal{S}'} \beta$ if $[\alpha]_{\mathcal{S}'} \subseteq [\beta]_{\mathcal{S}'}$, where $[\alpha]_{\mathcal{S}'}$ denotes the set of \mathcal{S}' -needed events of α .
 We write $\alpha \approx_{\mathcal{S}'} \beta$ iff $\alpha \trianglelefteq_{\mathcal{S}'} \beta$ and $\beta \trianglelefteq_{\mathcal{S}'} \alpha$, i.e., $[\alpha]_{\mathcal{S}'} = [\beta]_{\mathcal{S}'}$. We call $\trianglelefteq_{\mathcal{S}'}$ a *stable*
 13 *embedding* or *\mathcal{S}' -embedding*, and call $\approx_{\mathcal{S}'}$ a *stable equivalence* or *\mathcal{S}' -equivalence*.
 $\langle \alpha \rangle_{\mathcal{S}'}$ will denote the $\approx_{\mathcal{S}'}$ -equivalence class of α . We write $\langle \alpha \rangle_{\mathcal{S}'} \trianglelefteq_{\mathcal{S}'} \langle \beta \rangle_{\mathcal{S}'}$ if $\alpha \trianglelefteq_{\mathcal{S}'} \beta$.
 15 Finally, we define $\mathcal{L}^{\approx_{\mathcal{S}'}}(\mathcal{C}) = \mathcal{L}(\mathcal{C}) / \approx_{\mathcal{S}'} = \{ \langle \alpha \rangle_{\mathcal{S}'} \mid \alpha \in \mathcal{L}(\mathcal{C}) \}$ and $\mathcal{L}^{\trianglelefteq_{\mathcal{S}'}}(\mathcal{C}) = (\mathcal{L}^{\approx_{\mathcal{S}'}}(\mathcal{C}), \trianglelefteq_{\mathcal{S}'})$.

17 Note that $\approx_{\mathcal{S}'}$ does not satisfy [P0], although it does satisfy the other permutation
 axioms [P1]–[P3]. This means that for permutation equivalence ‘*progress in any direc-*
 19 *tion*’ is equally important, while for a stable equivalence $\approx_{\mathcal{S}'}$ only ‘*progress towards*
 \mathcal{S}' ’ matters. Actually, \mathcal{P} is a special case of $\approx_{\mathcal{S}'}$: the two coincide when \mathcal{C} is a PES
 21 and \mathcal{S}' is the set of all events.

Definition 62. Let \mathcal{S}' be a stable set of configurations in a DEES \mathcal{C} . The meet and
 23 join operations $\sqcup_{\mathcal{S}'}, \sqcap_{\mathcal{S}'}$ on $\mathcal{L}^{\trianglelefteq_{\mathcal{S}'}}(\mathcal{C})$ are defined by

- $\alpha \sqcup_{\mathcal{S}'} \beta = [\alpha]_{\mathcal{S}'} \cup [\beta]_{\mathcal{S}'}$ and $\langle \alpha \rangle_{\mathcal{S}'} \sqcup_{\mathcal{S}'} \langle \beta \rangle_{\mathcal{S}'} = \langle \alpha \sqcup_{\mathcal{S}'} \beta \rangle_{\mathcal{S}'}$;
- $\alpha \sqcap_{\mathcal{S}'} \beta = [\alpha]_{\mathcal{S}'} \cap [\beta]_{\mathcal{S}'}$ and $\langle \alpha \rangle_{\mathcal{S}'} \sqcap_{\mathcal{S}'} \langle \beta \rangle_{\mathcal{S}'} = \langle \alpha \sqcap_{\mathcal{S}'} \beta \rangle_{\mathcal{S}'}$.

It is immediate from the definition of $\approx_{\mathcal{S}'}$ that the above definition is correct.

27 **Lemma 63.** *Let $\beta \triangleright e$, in a DEES \mathcal{C} . Then $\beta \approx \beta \setminus [e] \triangleright$.*

Proof. If β is finite, then the lemma follows from Proposition 21. Otherwise, we have
 29 from Definition 57(1) and (2) and [E3] that $\beta \triangleright e \Rightarrow \exists \beta_0 \subseteq_{\text{fin}} \beta : \beta_0 \triangleright e \Rightarrow \beta_0 \triangleright [e] \triangleright$
 $\Rightarrow \beta \triangleright [e] \triangleright \Rightarrow \beta \approx \beta \setminus [e] \triangleright$. \square

31 **Lemma 64.** *Let \mathcal{S}' be a stable set of configurations in a DEES \mathcal{C} , and let e be
 \mathcal{S}' -needed. Then e is non-erasable: $\forall \alpha \in \mathcal{L}(\mathcal{C}) : \alpha \not\vdash e$.*

33 **Proof.** Suppose on the contrary that $\exists \alpha \in \mathcal{L}(\mathcal{C}) : \alpha \triangleright e$. We can assume that $\alpha \in \mathcal{L}_{\text{fin}}(\mathcal{C})$.
 By [E3] and [E2], $\alpha \setminus [e] \triangleright e$. But since e is \mathcal{S}' -needed, $\alpha \setminus [e] \triangleright \notin \mathcal{S}'$. Hence $\beta = \alpha_{\mathcal{S}'}$
 35 $\cup (\alpha \setminus [e] \triangleright) \in \mathcal{S}'$ by closure of \mathcal{S}' under inclusion. But $\beta \triangleright e$ by Lemma 58, and

1 $\beta \approx \beta \setminus [e] \geq$ by Lemma 63. Thus, by closure of \mathcal{S}' under permutation equivalence,
 $\beta \setminus [e] \geq \in \mathcal{S}'$, and $e \notin \beta \setminus [e] \geq$, contradicting \mathcal{S}' -neededness of e . \square

3 The following proposition shows that $\approx \subseteq \approx_{\mathcal{S}'}$, which agrees with our intuition
 5 that a concurrent semantics must not discriminate between permutation equivalent
 configurations/reductions.

Proposition 65. *Let α, β be configurations in a DEES \mathcal{C} , and let \mathcal{S}' be a stable set
 7 of configurations in \mathcal{C} . Then $\alpha \approx \beta$ implies $\alpha \approx_{\mathcal{S}'}$ β .*

Proof. By Definition 57(1), $\alpha \triangleright \beta \setminus \alpha$ and $\beta \triangleright \alpha \setminus \beta$. Hence, by Lemma 64, $[\alpha]_{\mathcal{S}'}, [\beta]_{\mathcal{S}'} \subseteq$
 9 $\alpha \cap \beta$, and by Theorem 59, $[\alpha]_{\mathcal{S}'} = \alpha \cap \alpha_{\mathcal{S}'} = (\alpha \cap \beta) \cap \alpha_{\mathcal{S}'} = \beta \cap \alpha_{\mathcal{S}'} = [\beta]_{\mathcal{S}'}$. \square

Theorem 66. *Let \mathcal{S}' be a stable set of configurations in a DEES \mathcal{C} . Then $\mathcal{L}^{\triangleleft_{\mathcal{S}'}}(\mathcal{C}) =$
 11 $(\mathcal{L}^{\approx_{\mathcal{S}'}}(\mathcal{C}), \triangleleft_{\mathcal{S}'})$ is a finitary prime algebraic complete lattice.*

Proof. Immediate from Definition 62 and Theorem 59, since the set of subsets of any
 13 non-empty set (in our case, $\alpha_{\mathcal{S}'}$) ordered by the subset relation is a finitary prime
 algebraic complete lattice with intersection and union as meet and join operations [37].
 15 \square

In an ASDRS \mathcal{R} , stability of a set \mathcal{S} of initial (finite or infinite) reductions can
 17 be defined via stability of the corresponding set \mathcal{S}' of configurations in $\mathcal{C}_{\mathcal{R}}$. For any
 initial reductions P, Q , and any stable set of reductions \mathcal{S} , in an ASDRS \mathcal{R} , we can
 19 define $P \triangleleft_{\mathcal{S}} Q$ iff $FAM_{\mathcal{S}}(P) \triangleleft_{\mathcal{S}'} FAM_{\mathcal{S}}(Q)$ in $\mathcal{C}_{\mathcal{R}}$, where $FAM_{\mathcal{S}}(P)$ denotes the set
 of \mathcal{S} -needed families contracted in P (and hence coincides with the set of \mathcal{S}' -needed
 21 events in the configuration corresponding to P). Further, we define $P \approx_{\mathcal{S}} Q$ iff $P \triangleleft_{\mathcal{S}} Q$
 and $Q \triangleleft_{\mathcal{S}} P$. We call $\triangleleft_{\mathcal{S}}$ and $\approx_{\mathcal{S}}$ a *stable embedding* or *\mathcal{S} -embedding* and a *stable*
 23 *equivalence* or *\mathcal{S} -equivalence*, respectively. $\langle P \rangle_S$ denotes the $\approx_{\mathcal{S}}$ -equivalence class of
 P . We write $\langle P \rangle_S \triangleleft_{\mathcal{S}} \langle Q \rangle_S$ if $P \triangleleft_{\mathcal{S}} Q$. Now, if $\mathcal{L}(\mathcal{R})$ is the set of (finite or infinite)
 25 initial reductions in \mathcal{R} , we can define $\mathcal{L}^{\approx_{\mathcal{S}}}(\mathcal{R}) = \mathcal{L}(\mathcal{R}) / \approx_{\mathcal{S}} = \{\langle P \rangle_S \mid P \in \mathcal{L}(\mathcal{R})\}$ and
 $\mathcal{L}^{\triangleleft_{\mathcal{S}}}(\mathcal{R}) = (\mathcal{L}^{\approx_{\mathcal{S}}}(\mathcal{R}), \triangleleft_{\mathcal{S}})$.

27 **Lemma 67.** *Let $\alpha \in \mathcal{L}(\mathcal{C})$. Then $[\alpha]_{\mathcal{S}'} \in \mathcal{L}(\mathcal{C})$. Moreover, $[\alpha]_{\mathcal{S}'}$ is standard, and any
 of its left-closed enumerations is a reduction.*

29 **Proof.** Let $e, e' \in \alpha$, let $e < e'$ and let e' be \mathcal{S}' -needed. We need to show that e is
 \mathcal{S}' -needed too. By Theorem 59, $e' \in \alpha_{\mathcal{S}'} \in \mathcal{L}(\mathcal{C})$, implying that $e \in \alpha_{\mathcal{S}'}$. Thus $[\alpha]_{\mathcal{S}'} \in$
 31 $\mathcal{L}(\mathcal{C})$ ($[\alpha]_{\mathcal{S}'}$ is left-closed). The rest is immediate from Lemma 64. \square

Theorem 68. *Let \mathcal{R} be an ASDRS and let $\mathcal{C} = \mathcal{C}_{\mathcal{R}}$ be its corresponding DEES.
 33 Further, let \mathcal{S}' be a set of configurations in \mathcal{C} , and let \mathcal{S} be the set of corre-
 sponding reductions in \mathcal{R} . Then $f : \langle P \rangle_S \rightarrow \langle FAM(P) \rangle_{\mathcal{S}'}$ is an isomorphism between
 35 $\mathcal{L}^{\triangleleft_{\mathcal{S}}}(\mathcal{R})$ and $\mathcal{L}^{\triangleleft_{\mathcal{S}'}}(\mathcal{C})$, with the inverse $f^{-1} : \langle \alpha \rangle_{\mathcal{S}'} \rightarrow \langle [[\alpha]_{\mathcal{S}'}] \rangle_S$.*

1 **Proof.** By Lemma 67, $[[\alpha]_{\mathcal{G}'}]$ and hence f^{-1} are well defined. The rest is immediate
from the definitions above. \square

3 Hence we can define on $\mathcal{L}^{\triangleleft_{\mathcal{G}}}(\mathcal{R})$ that $\langle P \rangle_S \sqcup_{\mathcal{G}} \langle Q \rangle_S = f^{-1}(f(\langle P \rangle_S) \sqcup_{\mathcal{G}'} f(\langle Q \rangle_S))$
and $\langle P \rangle_S \sqcap_{\mathcal{G}} \langle Q \rangle_S = f^{-1}(f(\langle P \rangle_S) \sqcap_{\mathcal{G}'} f(\langle Q \rangle_S))$, and we get from Theorem 66 that
5 $\mathcal{L}^{\triangleleft_{\mathcal{G}}}(\mathcal{R})$ with $\sqcup_{\mathcal{G}'}$ and $\sqcap_{\mathcal{G}'}$ as join and meet is a finitary prime algebraic domain.
It is not difficult to define stability of a set of reductions \mathcal{S} , as well as the relations
7 $\triangleleft_{\mathcal{G}}$, $\sqcup_{\mathcal{G}}$ and $\sqcap_{\mathcal{G}}$, on (corresponding equivalence classes of) reductions in \mathcal{R} directly.

8. Conclusions

9 We have established an equivalence between deterministic stable operational (AS-
DRS) and domain-theoretic (DEES) models of computation, and based on this, con-
11 structed a fully adequate event style concurrent semantics for orthogonal rewrite
systems. The correspondence between Event Structures and other models of concu-
13 rrency, as well as with denotational semantics, are well studied [37, 38], and we hope
that our results contribute to better understanding of the relationship between opera-
15 tional and denotational semantics of sequential and concurrent languages.

We think that our axiomatization of permutation equivalence is interesting on its
17 own right from the poset/lattice theoretic point of view. Definitions of permutation
equivalence without using a residual calculus have been studied before [25], but they
19 still use the syntax of the rewrite system essentially (axioms depend on the syntax).

Although we have restricted attention to orthogonal rewriting, it would be interesting
21 to extend our axiomatization to partial orders with conflicts, to enable modelling of
non-deterministic calculi as well.

23 Acknowledgements

We thank R. Kennaway for useful comments.

References

- 25 [1] A. Asperti, S. Guerrini, The Optimal Implementation of Functional Programming Languages, Cambridge
Tracts in Theoretical Computer Science, vol. 45, Cambridge University Press, Cambridge, 1998.
27 [2] A. Asperti, H.G. Mairson, Parallel beta reduction is not elementary recursive, in: Proc. 15th ACM
Symp. on Principles of Programming Languages, 1998, pp. 303–315.
29 [3] H.P. Barendregt, The Lambda Calculus, its Syntax and Semantics, North-Holland, Amsterdam, 1984.
[4] G. Berry, J.-J. Lévy, Minimal and optimal computations of recursive programs, J. Assoc. Comput. Mach.
31 26 (1979) 148–175.
[5] G. Boudol, Computational semantics of term rewriting systems, in: M. Nivat, J.C. Reynolds (Eds.),
33 Algebraic methods in semantics, Cambridge University Press, Cambridge, 1985,, pp. 169–236.
[6] G. Boudol, I. Castellani, A non-interleaving semantics for CCS based on proved transitions, Fund.
35 Inform. 11 (4) (1988) 433–452.
[7] D. Clark, J.R. Kennaway, Event structures and non-orthogonal term graph rewriting, Math. Struct.
37 Comput. Sci. 6 (1996) 545–578.

- 1 [8] A. Corradini, F. Gadduchi, U. Montanari, Relating two categorical models of term rewriting, in: Proc.
RTA'95, Lecture Notes in Computer Science, vol. 914, Springer, Berlin, 1995, pp. 225–240.
- 3 [9] J.R.W. Glauert, J.R. Kennaway, Z. Khasidashvili, Stable results and relative normalization, Journal of
Logic and Computation, 10(3) 2000. Special issue in: F. Kamareddine, J.W. Klop (Eds.), Type Theory
5 and Term Rewriting, Oxford University Press, Oxford.
- 7 [10] J.R.W. Glauert, Z. Khasidashvili, Relative normalization in orthogonal expression reduction systems, in:
Proc. CTRS'94, Lecture Notes in Computer Science, vol. 968, Springer, Berlin, 1994, pp. 144–165.
- 9 [11] J.R.W. Glauert, Z. Khasidashvili, Relative normalization in deterministic residual structures. in: Proc.
CAAP'96, Lecture Notes in Computer Science, vol. 1059, Springer, Berlin, 1996, pp. 180–195.
- 11 [12] G. Gonthier, J.-J. Lévy, P.-A. Melliès, An abstract standardisation theorem, in: Proc. 7th IEEE Symp.
on Logic in Computer Science, 1992, pp. 72–81.
- 13 [13] R.J. Hindley, An abstract form of the Church-Rosser theorem I, J. Symbolic Logic 34 (4) (1969)
545–560.
- 15 [14] G. Huet, J.-J. Lévy, Computations in orthogonal rewriting systems, in: J.-L. Lassez, G. Plotkin (Eds.),
Computational Logic, Essays in Honor of Alan Robinson, MIT Press, Cambridge, MA, 1991, pp. 394–
443.
- 17 [15] V. Kathail, Optimal interpreters for lambda-calculus based functional languages, Ph.D. Thesis, MIT,
1990.
- 19 [16] J.R. Kennaway, J.W. Klop, M.R. Sleep, 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
21 and Practice, Wiley, New York, 1993, pp. 141–156.
- 23 [17] Z. Khasidashvili, 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, pp.
135–149.
- 25 [18] Z. Khasidashvili, J.R.W. Glauert, Zig-zag, extraction and separable families in non-duplicating stable
deterministic residual structures, Tech. Report IR-420, Free University, 1997.
- 27 [19] Z. Khasidashvili, 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,
29 pp. 269–278.
- 31 [20] Z. Khasidashvili, J.R.W. Glauert, An abstract concept of optimal implementation, Tech. Report
SYS-C97-01, UEA Norwich, 1997.
- 33 [21] J.W. Klop, Combinatory Reduction Systems, Mathematical Centre Tracts, vol. 127, Amsterdam, 1980.
- 35 [22] J.W. Klop, Term rewriting systems, in: S. Abramsky, D. Gabbay, T. Maibaum (Eds.), Handbook of
Logic in Computer Science, vol. 2I, Oxford University Press, Oxford, 1992, pp. 1–116.
- 37 [23] J. Lamping, An algorithm for optimal lambda calculus reduction, in: Proc. 7th ACM Symp. on Principles
of Programming Languages, 1990, pp. 6–30.
- 39 [24] C. Laneve, Distributive evaluations of λ -calculus, Fund. Inform. 20 (4) (1994) 333–352.
- 41 [25] C. Laneve, U. Montanari, Axiomatizing permutation equivalence, Math. Struct. Comput. Sci. 6 (1994)
219–249.
- 43 [26] J.-J. Lévy, Réductions correctes et optimales dans le lambda-calcul, Thèse de l'Université de Paris VII,
1978.
- 45 [27] J.-J. Lévy, Optimal reductions in the Lambda-calculus, in: J.P. Seldin, J.R. Hindley (Eds.), To H.B.
Curry: Essays on Combinatory Logic, Lambda-calculus and Formalism, Academic Press, New York,
1980, pp. 159–192.
- 47 [28] 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, pp. 255–269.
- 49 [29] P.-A. Melliès, Description Abstraite des Systèmes de Réécriture, Thèse de l'Université Paris 7, 1996.
- 51 [30] P.-A. Melliès, A factorization theorem in rewriting theory, in: Proc. CTCS'97, Lecture Notes in
Computer Science, vol. 1290, Springer, Berlin, 1997, pp. 49–68.
- 53 [31] R. Milner, Communication and Concurrency, Prentice-Hall, Englewood Cliffs, NJ, 1989.
- 55 [32] M. Nielsen, G. Plotkin, G. Winskel, Petri nets, event structures and domains, Part 1, Theoret. Comput.
Sci. 13 (1981) 85–108.
- 57 [33] D. Sangiorgi, Expressing mobility in process algebras: first-order and higher-order paradigms, Ph.D.
Thesis, Edinburgh University, 1993.
- [34] G. Schied, On relating rewrite systems and graph grammars to event structures, in: Proc. GTCS'94,
Lecture Notes in Computer Science, vol. 776, Springer, Berlin, 1994, pp. 326–340.
- [35] E.W. Stark, Concurrent transition systems, Theoret. Comput. Sci. 64 (3) (1989) 221–269.

- 1 [36] G. Winskel, Events in computation, Ph.D. Thesis, Edinburgh University, 1980.
- 3 [37] 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,
5 Berlin, 1989, pp. 364–397 (extended version appears as Cambridge University Computer Laboratory
Report no. 95, 1986).
- 7 [38] G. Winskel, M. Nielsen, Models for concurrency, in: S. Abramsky, D. Gabbay, T. Maibaum (Eds.),
Handbook of Logic in Computer Science, vol. 4, Oxford University Press, Oxford, 1995, pp. 1–148.

UNCORRECTED PROOF