Journal of Logical and Algebraic Methods in Programming 104 (2019) 16–59

Contents lists available at ScienceDirect

Journal of Logical and Algebraic Methods in Programming

www.elsevier.com/locate/jlamp

Towards a categorical representation of reversible event structures

Eva Graversen *, Iain Phillips, Nobuko Yoshida

Imperial College London, UK

A R T I C L E I N F O

Article history:
Received 15 August 2017
Received in revised form 10 December 2018
Accepted 7 January 2019
Available online 17 January 2019

Keywords:
Reversible computation
Reversible event structures
Category theory

A B S T R A C T

We study categories for reversible computing, focussing on reversible forms of event structures. Event structures are a well-established model of true concurrency. There exist a number of forms of event structures, including prime event structures, asymmetric event structures, and general event structures. More recently, reversible forms of these types of event structure have been defined. We formulate corresponding categories and functors between them. We show that products and coproducts exist in many cases.

We define stable reversible general event structures and stable configuration systems, and we obtain an isomorphism between the subcategory of the former in normal form and the finitely enabled subcategory of the latter.

In most work on reversible computing, including reversible process calculi, a causality condition is posited, meaning that the cause of an event may not be reversed before the event itself. Since reversible event structures are not assumed to be causal in general, we also define causal subcategories of these event structures.

1. Introduction

Event structures are a well-known model of true concurrency. They consist of events and relations on events, describing the causes of events and conflict between events. Since the original introduction of prime event structures [16], related notions such as general event structures [25], often called simply event structures, which replace conflict with a set of consistent sets of events and causation with sets of enabling events, and asymmetric event structures [3], which make the conflict asymmetric, have been introduced. Winskel [24] defined a category of event structures, and used this to define event structure semantics of CCS.

Reversible process calculi are a well-studied field [14,8,9,17,6,13]. When considering the semantics of reversible processes, the ability to reverse events leads to finer distinctions of a true concurrency character [18]. For instance, using CCS notation, the processes $a \mid b$ and $a.b + b.a$ are equivalent under interleaving semantics; however in a reversible setting we can distinguish them by noting that $a \mid b$ allows us to perform (an event labelled by) a followed by b and then to reverse a, which is impossible for $a.b + b.a$. This motivates the study of *reversible event structures*.

Reversible versions of various kinds of event structures were introduced in [19,21]. Our aim here is to interpret these as objects in appropriate categories and study functors between them (as a foundation for using reversible event structures to give true concurrency semantics of reversible process calculi). Hitherto few reversible frameworks have been defined

* Corresponding author.
E-mail addresses: e.graversen16@imperial.ac.uk (E. Graversen), i.phillips@imperial.ac.uk (I. Phillips), n.yoshida@imperial.ac.uk (N. Yoshida).

https://doi.org/10.1016/j.jlamp.2019.01.001
2352-2208/

categorically, though [10] used category theory to describe the relationship between RCCS processes and their histories, and [5] used dagger categories to define a reversible process calculus called Π.

We formulate categories for the reversible prime event structures (RPESs) and reversible asymmetric event structures (RAESs) of [19] and for the reversible (general) event structures (RESs) of [21]. We define morphisms for each category and functors between them, which in some cases form adjunctions. We construct coproducts, and, in the case of RESs and a subcategory of RPESs (see Remark 4.22), products.

In the context of event structures and other models of concurrency, morphisms act as partial synchronisations. Event structure morphisms are partial functions mapping events from one event structure to events from another in such a way that (1) any acceptable configuration of events from the first structure is also an acceptable configuration in the second, and (2) if an event e_1 from the first structure is enabled in such a configuration and mapped to an event e_2 in the second structure, then applying the morphism to the configuration gives us a configuration in which e_2 is enabled. When two functors $F : B \to A$ and $G : A \to B$ between a category of more powerful event structures A and a category of less powerful event structures B form an adjunction, we can say that G is an approximation which underestimates or overestimates the capabilities of the event structure described by the more powerful framework A when translating to B. We anticipate that products and coproducts will be useful when it comes to modelling reversible process calculi, as they were used by [24] to model parallel composition and choice respectively in the forward-only setting.

Although no functor is known from asymmetric event structures to general event structures such that the general event structure maps to the same domain as the asymmetric, we define a functor from reversible asymmetric event structures to reversible general event structures and show a correspondence between them (Theorem 6.23).

RPESs and RAESs differ from forward-only prime and asymmetric event structures by adding not only the ability for events to reverse, but also having a causality and prevention relation describing when events are allowed to reverse. RESs on the other hand modify their enablings, adding asymmetric conflict specific to each enabling in the form of a preventing set. Since the preventing set is specific to each enabling, RESs can have multiple different asymmetric conflicts associated with an event depending on the enabling being used. This is similar to how forward-only general event structures can have multiple different sets of causes, or enabling sets, associated with an event, unlike prime and asymmetric event structures, which can only have one.

We believe these variants of event structures may all prove useful for describing different reversible processes, as we shall illustrate in Examples 1.1, 1.2, and 1.3. Example 1.3 is related to the parallel switch of Example 1.1.7 of [25].

Having functors between the different variants of event structures improves our ability to compare systems modelled by different kinds of event structures. Examples 1.1 and 1.3 show reversible prime and general event structures being used to model CCS and π-calculus processes respectively. Having a functor which maps the RPES created by the CCS process to an RES allows us to compare the processes (e.g. by defining morphisms between their corresponding RESs). We can also use the product or coproduct of the RESs to get a parallel composition or choice between the processes if they are intended to be part of a larger system.

Example 1.1 *(Using an RPES to model reversible CCS).* The simple CCS process $a \mid \overline{a}.b$ can be modelled by an RPES which splits the b into two events, one caused by $\overline{a}$ and one caused by τ, wherein a and $\overline{a}$ are both in conflict with τ. We can also say that both b's prevent τ and $\overline{a}$ from reversing before they do. We return to this in Example 4.2.

Example 1.2 *(Using an RAES to model an error [19]).* Suppose we wish to model a computation wherein a series of events, e_0,e_1,e_2,... happens in a sequential fashion unless interrupted by an error event, which causes the computation to undo all events in reverse order and start over. We would need the error event to prevent the computation from continuing. However, we would not want any events from the computation to prevent the error from happening as we would have if we say error is in conflict with e_i in an RPES. To model this we need asymmetric conflict: error prevents e_i. We return to this in Example 5.2.

Example 1.3 *(Using an RES to model π-calculus [6]).* Consider the π-calculus process $(\nu n)(\overline{a}\langle n\rangle \mid \overline{b}\langle n\rangle \mid c(x))$, wherein a bound name n can be extruded on either channel a or b after which it can be received by c. This creates an action with two possible causes, which are not mutually exclusive. If c receives n after both outputs have been performed then either output could have caused the input, since either output could have been what caused n to become free and therefore available for c to receive. Hence one can argue that to describe this process with a reversible event structure, we would need $\overline{a}\langle n\rangle$ or $\overline{b}\langle n\rangle$ to be able to cause $c(n)$. Also we would want either output to be able to reverse after $c(n)$ has been performed as long as the other output has also been performed and not reversed. We return to this in Example 6.2.

We also create a category of *configuration systems* (CSs) [19]. CSs are a model of concurrency intended to serve a similar purpose as domains do for the forward-only event structures, letting the various kinds of reversible event structures be translated into one formalism. A CS consists of a set of events, a set of configurations of events, and transitions between them. We show that, just as domains can be modelled as event structures, so a subclass of CSs which we call *finitely enabled* can be modelled as RESs. We prove that our functors between RESs and finitely enabled CSs form an adjunction (Theorem 6.17). Furthermore they are inverses for finitely enabled CSs under certain conditions (Theorem 6.16). Having these mappings in both directions is very useful, as modelling a system as a CS can often be more intuitive than an event structure, especially an RES with many enablings.

We further define a subcategory of the RESs which are *stable* (SRESs), meaning that the causes of an event cannot be ambiguous. While Example 1.3 required the use of a non-stable RES, Example 1.4 below describes a case where we would want to use an SRES to describe a reversible computation. We also define stable CSs, and functors between them and stable RESs.

Example 1.4 *(Using an SRES to model multiple errors).* Consider the scenario in Example 1.2, wherein a computation can be interrupted and forced to restart by an error event. We now modify it so that we have multiple error events, only one of which can happen at a time. Then we need to be able to say that an event e_i in the computation can be reversed if one of the mutually exclusive error events has happened. We cannot achieve this with RAESs, as having multiple error events cause the same event to reverse would imply all the error events need to have occurred before we can reverse. However with SRESs we can realise the desired behaviour by creating a separate enabling for each error event. Since only one error event can exist in any given configuration, this will be stable. We return to this in Example 7.2.

We present a definition of normal form for RESs, which ensures that we only have the minimum enablings necessary to describe the corresponding CS. This in particular gives us mappings which are full inverses between stable finitely enabled CSs and stable RESs in normal form (Proposition 7.13). We also define a normaliser functor (Definition 7.17), which maps RESs to RESs in normal form. Some non-stable RESs become stable when normalised. Since there only exists one stable RES in normal form which describes any given stable CS, normalising two stable RESs which map to the same CS also gives us the same result. This makes normalising a useful tool for comparing stable RESs and to a lesser degree non-stable RESs.

With a few exceptions [20,22], reversible process calculi have always adopted *causal* reversibility, where an action can be reversed if and only if all the actions caused by it have been reversed first. By contrast, the reversible event structures of [19,21] that we consider here allow non-causal reversibility, inspired by bonding in biochemical processes. Hence causal subcategories are of interest.

We define subcategories of RESs which are (1) *cause-respecting*, meaning that no action can be reversed unless all the actions caused by it have been reversed first [19], which can be seen as a safety property for causal reversibility, and (2) *causal*, which means both cause-respecting and also that events whose consequences have all been reversed can themselves be reversed (a liveness property). Consider again Examples 1.1 and 1.2. The RPES in Example 1.1 is causal, but the error event in Example 1.2 makes the RAES non-causal, as the most recent e_i event cannot reverse, despite not having caused any other events, unless the error event is also present in the configuration.

Our longer-term aim is to formulate event structure semantics for reversible process calculi. As a first step towards this, we have in subsequent work [12] defined event structure semantics of CCSK using causal reversible bundle event structures, and of CCSK with rollback using non-causal reversible extended bundle event structures. Bundle event structures are more expressive than prime event structures, but less expressive than general event structures. The causal reversible bundle event structures without cycles in their causation which we used to describe the semantics of CCSK could potentially have been expressed as causal reversible prime event structures, given that products exist in the latter (Remark 4.22), although the non-causal reversible extended bundle event structures we used to model rollback could not.

Other models of concurrency have also been used to define semantics of reversible process calculi. The reversible π-calculus has been modelled using rigid families [7], and RCCS has been modelled using configuration structures [1]. Both of these took advantage of the fact that categorical definitions of rigid families and configuration structures exist, as they used the products and coproducts to model parallel composition and choice. However, while both rigid families and configuration structures can model causality, since neither was designed to be reversible, they cannot model situations such as the ones seen in Examples 1.2 and 1.4, where the reverse transitions are not the exact inverse of the forwards transitions.

Structure of the paper Section 2 reviews forwards-only event structures, on which our reversible event structures are based. Section 3 describes the category of configuration systems, which we use to model reversible event structures. Section 4 describes the category of the reversible variant of prime event structures, while Sections 5 and 6 cover reversible asymmetric and reversible general event structures. Section 7 investigates stable variants of general reversible event structures and configuration systems. Finally Section 8 describes cause-respecting and causal variants of general reversible event structures and configuration systems. Fig. 1 shows an overview of the categories and functors introduced in this paper.

This paper is revised and extended from the preliminary workshop paper [11]. The differences can be summarised as follows:

- We present an additional mapping from finitely enabled configuration systems to reversible event structures in Definition 6.14, which acts as a left adjoint to the mapping from reversible event structures to configuration systems (Theorem 6.17). It also functions as an inverse for a different group of configuration systems (Theorem 6.16).

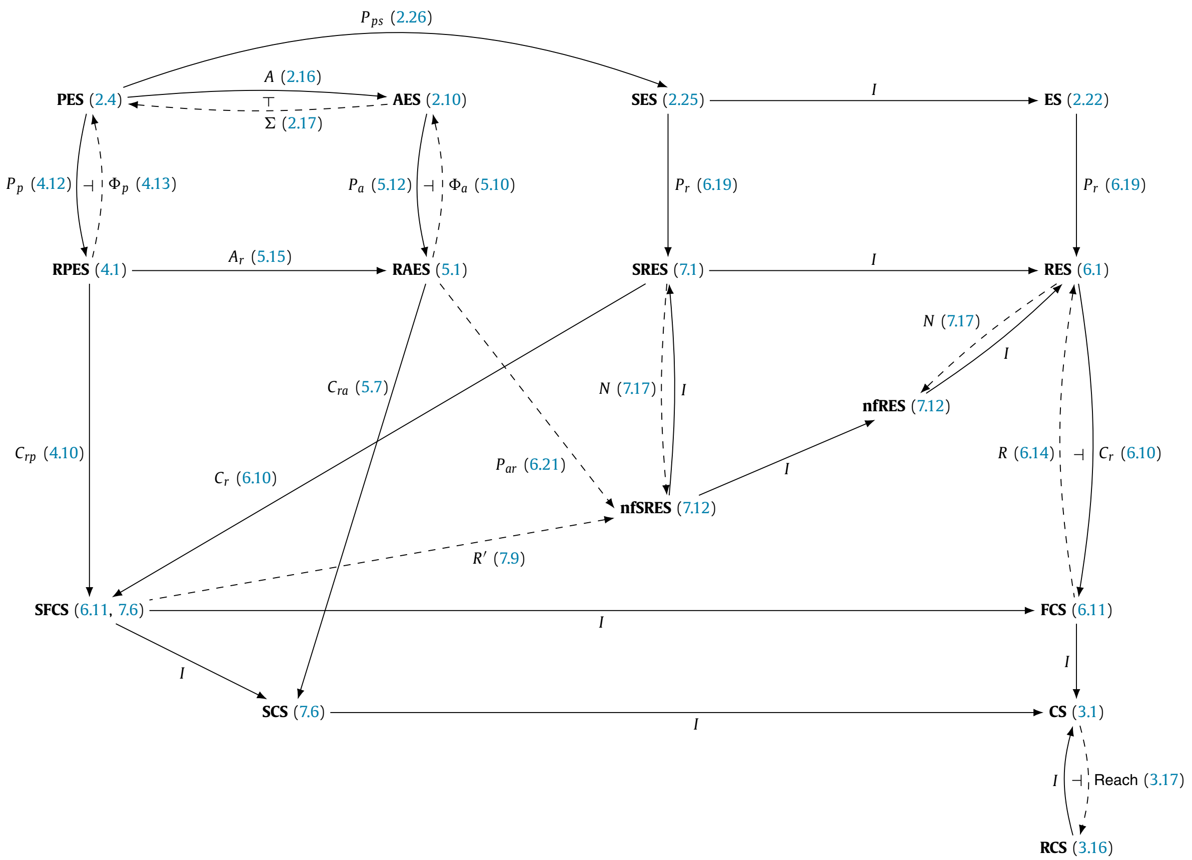


Fig. 1. Categories of event structures and functors between them, with the definitions where they are introduced. We use the prefix **S** to denote the stable variants, **nf** to denote the RESs in normal form, and **F** to denote the finitely enabled CSs. We use dashed lines to indicate functors which do not commute with the rest of the diagram and I to represent inclusions.

- We correct Definitions 7.9 and 7.6 and Proposition 7.13 because we found that some stable RESs did not correspond to stable CSs. We also move Definition 7.9 to Section 7, as the new functor is much better suited for dealing with non-stable cases, while the old one preserves stability (Propositions 7.10 and 7.11).
- We have added the notion of normal form for RESs mentioned above.
- We present revised definitions of causal and cause-respecting reversible event structures and configuration systems.
- We define reachable configuration systems as ones in which any configuration is reachable by a series of transitions from the empty configuration. We define a functor restricting CSs to their reachable parts, and show that it forms a right adjoint of an inclusion functor. Based on this we define reachably cause-respecting (resp. reachably causal) CSs as ones where the reachable part is cause-respecting (resp. causal).
- We provide proofs of all results.

2. Forwards-only event structures

Before describing the different categories of reversible event structures, we review the categories of forward-only event structures and domains, an overview of which can be found in Fig. 2. The only new work in this section is extending the mapping from asymmetric event structures to prime event structures into a functor, and proving that it forms the left adjunction of an established functor from prime event structures to asymmetric event structures (Proposition 2.20).

2.1. Domains

Forward-only event structures are based on, and map into, a kind of partial order called domains. To define domains we first need some notation on partially ordered sets $\langle D, \sqsubseteq \rangle$ from [25]:

Given $x \in D$, $\downarrow x = \{x' \in D \mid x' \sqsubseteq x\}$.

We write $\uparrow X$ if $X \subseteq D$ is *compatible*, meaning there exists a $y \in D$ such that for all $x \in X$, $x \sqsubseteq y$.

$X \subseteq D$ is *pairwise compatible* if for all $x, y \in X$, $\uparrow\{x, y\}$.

$X \subseteq D$ is *directed* if for any $x, y \in X$ there exists $z \in X$ such that $x \sqsubseteq z$ and $y \sqsubseteq z$.

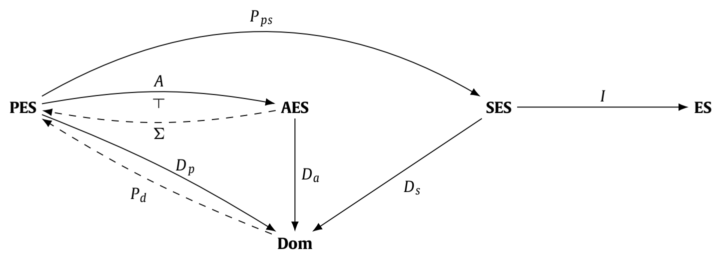


Fig. 2. Categories of forward-only event structures and functors between them: PES were introduced in [16], and defined categorically along with **Dom**, **SES**, **ES**, D_p, P_d, P_{ps}, and D_s in [25], **AES**, A, and D_a were introduced in [3], and Σ as a mapping in [19]. The adjunction between A and Σ is new to this work. Dashed lines indicate functors which do not commute with the rest of the diagram.

$\langle D, \sqsubseteq \rangle$ is a *complete partial order* (CPO) if for any directed subset $X \subseteq D$, there exists a least upper bound $\bigsqcup X \in D$.
Element $y \in D$ is *compact* if for any directed set $X \subseteq D$, if $y \sqsubseteq \bigsqcup X$ then there exists an $x \in X$ such that $y \sqsubseteq x$.
$\mathsf{K}(D)$ is the set of compact elements of D.
A CPO $\langle D, \sqsubseteq \rangle$ is *finitary* if for all $x \in \mathsf{K}(D)$, $\downarrow x$ is finite.
A partial order $\langle D, \sqsubseteq \rangle$ is *coherent* if for all pairwise compatible $X \subseteq D$, there exists a least upper bound $\bigsqcup X \in D$.
A *complete prime* of D is an element $y \in D$ such that for any compatible $X \subseteq D$, if $y \sqsubseteq \bigsqcup X$ then there exists an $x \in X$ such that $y \sqsubseteq x$.
The set of complete primes of D is denoted $Pr(D)$.
Given $x \in D$, $Pr(x) = \downarrow x \cap Pr(D)$.
A partial order $\langle D, \sqsubseteq \rangle$ is *prime algebraic* if for any $x \in D$, $x = \bigsqcup Pr(x)$.

Definition 2.1. *Domains* are coherent, prime algebraic, finitary partial orders.

Complete primes are considered to be atomic *events*, which cannot be expressed as the upper bound of other elements, while non-prime elements x of domains are the states reached when the events of which x is the least upper bound have happened.

To define the category **Dom**, we need a notion of domain morphism, which we present in Definition 2.2.

Definition 2.2 *(Domain Morphism [25]).* Let $\langle D_0, \sqsubseteq_0 \rangle$ and $\langle D_1, \sqsubseteq_1 \rangle$ be domains. A domain morphism is a function $f : D_0 \to D_1$ such that:

1. for all $x_0, y_0 \in D_0$, if $x \preceq_0 y$ then $f(x) \preceq_1 f(y)$ where $\prec$ is defined as:

$$x \prec x' \text{ if } x \sqsubseteq x' \text{ and } \forall x'' \in D.\ (x \sqsubseteq x'' \sqsubseteq x' \text{ implies } x = x'' \text{ or } x' = x'')\,;$$

2. for all $X \subseteq D_0$, if X is pairwise compatible then $f(\bigsqcup X) = \bigsqcup f(X)$;
3. for all $X \subseteq D_0$, if $X \neq \emptyset$ and is compatible then $f(\bigsqcap X) = \bigsqcap f(X)$.

We also define prime intervals in Definition 2.3 for use in future mappings.

Definition 2.3 *(Prime Interval [25]).* Let $\langle D, \sqsubseteq \rangle$ be a domain. A prime interval is a pair $[x, x']$ of elements of D such that $x \prec x'$. We say $[x, x'] \leq [y, y']$ if $x = x' \sqcap y$ and $x \sqcup y = y'$, and $\sim$ is the transitive and symmetric closure of $\leq$.

We furthermore define the mapping $[x, x']_\sim \mapsto p$, where p is the unique element in $Pr(x') \setminus Pr(x)$, which is an isomorphism between the $\sim$-classes of prime intervals of $\langle D, \subseteq \rangle$, and the complete primes of D, with the inverse function $p \mapsto [\bigsqcap\{x \in D \mid x \sqsubset p\}, p]_\sim$.

2.2. Prime event structures

We now recall the definition of *prime event structures*.

Definition 2.4 *(Prime Event Structure [16]).* A prime event structure (PES) is a triple $\mathcal{E} = (E, <, \#)$, where E is the set of events, and *causality*, $<$, and *conflict*, $\#$, are binary relations on E such that

1. $\#$ is irreflexive and symmetric;
2. $<$ is an irreflexive partial order such that for every $e \in E$, $\{e' \mid e' < e\}$ is finite;
3. $\#$ is hereditary with respect to $<$, i.e. for all $e, e', e'' \in E$, if $e \# e'$ and $e < e''$ then $e'' \# e'$.

(a) PES $\mathcal{E}$ (b) Domain of PES $\mathcal{E}$

Fig. 3. Example of a PES and corresponding domain.

A PES consists of a set of events, and causality and conflict relations describing when these events can occur. If $e < e'$ then e' cannot happen unless e has already happened. Moreover if $e \# e'$ then e and e' cannot both occur in the same computation. No event is allowed to have an infinite number of causes, as every event should be reachable in a finite number of steps. Furthermore, # is hereditary with respect to $<$, since if e and e' never appear in the same configuration, and e'' cannot appear in a configuration that does not contain e, then e'' and e' will never appear in the same configuration. This condition, along with # being irreflexive, also means that two events cannot be in conflict if they cause the same event.

Definition 2.5 *(Configuration [16]).* For any PES $\mathcal{E} = (E, <, \#)$, we say that $X \subseteq E$ is a *configuration* of $\mathcal{E}$ if X is left-closed under $<$ and conflict-free, meaning no $e, e' \in X$ exist, such that $e \# e'$. We use $\mathsf{Conf}(\mathcal{E})$ to denote the set of all configurations of $\mathcal{E}$.

We recall the definition of a PES morphism in Definition 2.6, giving us the category **PES**. Morphisms on event structures can be seen as a sort of synchronisation between the two structures, where if X is a configuration then $f(X)$ is too, and two events e and e' can only synchronise with the same image event $f(e) = f(e')$ if they are in conflict.

Definition 2.6 *(PES morphism [25]).* Let $\mathcal{E}_0 = (E_0, <_0, \#_0)$ and $\mathcal{E}_1 = (E_1, <_1, \#_1)$ be PESs. A morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$ is a partial function $f : E_0 \to E_1$ such that

1. for all $e \in E_0$, if $f(e) \neq \bot$ then $\{e_1 \mid e_1 <_1 f(e)\} \subseteq \{f(e') \mid e' <_0 e\}$;
2. for all $e, e' \in E_0$, if $f(e) \neq \bot \neq f(e')$ and $f(e) \#_1 f(e')$, then $e \#_0 e'$;
3. for all $e, e' \in E_0$, if $f(e) \neq \bot \neq f(e')$, $f(e) = f(e')$, and $e \neq e'$, then $e \#_0 e'$.

We can now describe functors between **Dom** and **PES**.

Definition 2.7 *(From **PES** to **Dom** [25]).* The functor $D_p : \mathbf{PES} \to \mathbf{Dom}$ is defined as:

1. $D_p(\mathcal{E}) = (\mathsf{Conf}(\mathcal{E}), \subseteq)$;
2. $D_p(f) = f'$ where $f'(X) = \{f(e) \mid e \in X\}$.

Definition 2.8 *(From **Dom** to **PES** [25]).* The functor $P_d : \mathbf{Dom} \to \mathbf{PES}$ is defined as:

1. $P_d(\langle D, \sqsubseteq \rangle) = \langle Pr(D), <, \# \rangle$ where $p < p'$ if $p \sqsubset p'$ and $p \# p'$ if not $p \uparrow p'$;
2. $P_d(f) = f'$ where

$$f(p_0) = \begin{cases} p_1 & \text{if } p_0 \mapsto [x_0, x_0']_\sim,\ f(x_0) \leq f(x_0') \text{ and } [f(x_0), f(x_0')]_\sim \mapsto p_1 \\ \bot & \text{otherwise.} \end{cases}$$

Example 2.9 shows a PES modelled as a domain.

Example 2.9. The PES $\mathcal{E}$, shown in Fig. 3a, with events a, b, c where $a < b$, $a < c$, and $b \# c$, has configurations $\emptyset$, $\{a\}$, $\{a, b\}$, and $\{a, c\}$, and therefore $D_p(\mathcal{E}) = \mathcal{D}_{\mathcal{E}}$ is the domain seen in Fig. 3b.

2.3. Asymmetric event structures

Asymmetric event structures resemble prime event structures, with the difference being that the conflict relation $e \rhd e'$[1] is asymmetric, so that rather than e and e' being unable to coexist in a configuration, e' cannot be added to a configuration

[1] [3] uses the notation $e \nearrow e'$ to denote the inverse of this.

that contains e. The inverse of this relation, written $e' \lhd e$, may also be seen as a relation of precedence or weak causality, where if $e' \lhd e$ (i.e. $e \rhd e'$) and both events are in a configuration then e' must have been added first. Definition 2.10 describes asymmetric event structures.

Definition 2.10 *(Asymmetric Event Structure [3]).* An asymmetric event structure (AES) is a triple $\mathcal{E} = (E, <, \lhd)$ where E is a set of events, $\lhd$ is the precedence relation, $<$ is the causality relation, which is an irreflexive partial order, and for all $e, e' \in E$:

1. $\{e'' \in E \mid e'' < e\}$ is finite and contains no $\lhd$ cycles;
2. if $e < e'$ then $e \lhd e'$;
3. if $e \# e'$ and $e < e''$ then $e'' \# e'$, where $\# = \lhd \cap \rhd$.

The configurations of an AES are defined in Definition 2.11. It replaces the conflict-freeness condition in Definition 2.5 with requiring $\lhd$ to be well-founded, since the configuration would not otherwise be reachable in a finite number of steps, as $e \lhd e'$ means e must have been performed before e' if they are in the same configuration. This condition also prevents cycles of asymmetric conflict, which would make the configuration unreachable. Conditions 1 and 2 ensure that the partial order of an AES's configurations will be finitary, which is necessary for mapping AESs to domains.

Definition 2.11 *(AES Configuration [3]).* Given an AES $\mathcal{E} = (E, <, \lhd)$, A configuration of $\mathcal{E}$ is a set $X \subseteq E$ such that

1. $\lhd$ is well-founded on X;
2. for all $e \in X$, $\{e' \in X \mid e' \lhd e\}$ is finite;
3. X is left-closed with respect to $<$.

The set of all configurations of $\mathcal{E}$ is denoted by $\mathsf{Conf}(\mathcal{E})$.

We define a notion of morphism in Definition 2.12, giving us the category **AES**. This works very similarly to PES morphisms, again creating a sort of synchronisation between two AESs.

Definition 2.12 *(AES morphism [3]).* Let $\mathcal{E}_0 = (E_0, <_0, \lhd_0)$ and $\mathcal{E}_1 = (E_1, <_1, \lhd_1)$ be AESs. A morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$ is a partial function $f : E_0 \to E_1$ such that

1. for all $e \in E_0$, if $f(e) \neq \bot$ then $\{e_1 \mid e_1 <_1 f(e)\} \subseteq \{f(e') \mid e' <_0 e\}$;
2. for all $e, e' \in E_0$, if $f(e) \neq \bot \neq f(e')$ and $f(e) \rhd_1 f(e')$, then $e \rhd_0 e'$;
3. for all $e, e' \in E_0$, if $f(e) \neq \bot \neq f(e')$, $f(e) = f(e')$, and $e \neq e'$, then $e \rhd_0 e'$.

We define a functor from **AES** to **Dom** in Definition 2.14, but to do this we first need an ordering of configurations, as defined in Definition 2.13. Like the functor from **PES** to **Dom**, the functor from **AES** to **Dom** maps event structures to their set of configurations; however, instead of ordering configurations by set inclusion, it orders them by the extension ordering, which allows new events to be added to a configuration only if they are not prevented by any of the existing events.

Definition 2.13 *($\sqsubseteq$ on configurations [3]).* Given an AES $\mathcal{E} = (E, <, \lhd)$, and configurations $X, X' \in \mathsf{Conf}(\mathcal{E})$, we say X' *extends* X, written $X \sqsubseteq X'$ if:

1. $X \subseteq X'$;
2. no $e \in X$ and $e' \in X' \setminus X$ exist, such that $e' \lhd e$.

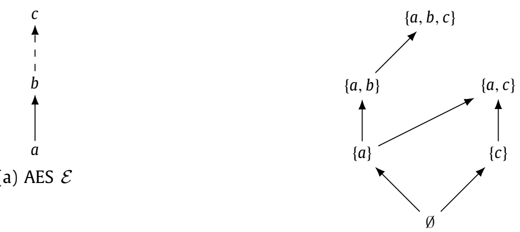
Definition 2.14 *(From **AES** to **Dom** [3]).* The functor $D_a : \mathbf{AES} \to \mathbf{Dom}$ is defined as:

1. $D_a(\mathcal{E}) = \langle \mathsf{Conf}(\mathcal{E}), \sqsubseteq \rangle$;
2. given an AES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$, $D_a(f) = f^*$ where for all $X \in \mathsf{Conf}(\mathcal{E}_0)$, $f^*(X) = \{f(e) \mid e \in X\}$.

Example 2.15 shows an AES modelled as a domain.

Example 2.15. Let the AES $\mathcal{E} = (E, <, \lhd)$ be given by $E = \{a, b, c\}$ and $a < b$ and $b \lhd c$. See Fig. 4a, where we use dotted arrows for weak causality $\lhd$. Then $\mathcal{E}$ has configurations $\emptyset$, $\{a\}$, $\{c\}$, $\{a, b\}$, $\{a, c\}$, and $\{a, b, c\}$, and therefore $D_a(\mathcal{E})$ is the domain seen in Fig. 4b.

We define functors between **AES** and **PES**.



(a) AES $\mathcal{E}$

(b) Domain of AES $\mathcal{E}$

Fig. 4. Example of an AES and the corresponding domain.

Definition 2.16 *(From **PES** to **AES** [3]).* The functor $A : \mathbf{PES} \to \mathbf{AES}$ is defined by:

1. $A((E, <, \#)) = (E, <, < \cup \#)$;
2. $A(f) = f$.

The following functor from **AES** to **PES** is based on a mapping given in [19].

Definition 2.17 *(From **AES** to **PES**).* The functor $\Sigma : \mathbf{AES} \to \mathbf{PES}$ is defined by:

1. $\Sigma((E, <, \lhd)) = (E, <, \lhd \cap \rhd)$;
2. $\Sigma(f) = f$.

Proposition 2.18. *$\Sigma : \mathbf{AES} \to \mathbf{PES}$ is a functor.*

Proof. [19] showed that Σ maps AESs to PESs, so we need to show that for an AES morphism $f : \mathcal{E} \to \mathcal{E}'$, $\Sigma(f) : \Sigma(\mathcal{E}) \to \Sigma(\mathcal{E}')$ is a PES morphism satisfying the conditions of Definition 2.6:

1. This is condition one of an AES morphism.
2. If $\Sigma(f)(e) \#' \Sigma(f)(e')$ then $f(e) \lhd' f(e')$ and $f(e) \rhd' f(e')$, which by definition of an AES morphism implies $e \lhd e'$ and $e \rhd e'$. Hence $e \# e'$ in $\Sigma(\mathcal{E})$.
3. If $\Sigma(f)(e) = \Sigma(f)(e')$ and $e \neq e'$ then, again by definition of an AES morphism, $e \lhd e'$ and $e \rhd e'$.

Having established that Σ maps between the categories, we show it to be a functor. The proof is split in three parts:

$\Sigma(f : \mathcal{E} \to \mathcal{E}') = \Sigma(f) : \Sigma(\mathcal{E}) \to \Sigma(\mathcal{E}')$:
Obvious since $\Sigma(f) = f$, $\Sigma(E) = E$, and $\Sigma(E') = E'$.

$\Sigma(1_{\mathcal{E}}) = 1_{\Sigma(\mathcal{E})}$:
Obvious since the identity function for all PES and AES objects is $f(e) = e$.

$\Sigma(f \circ f') = \Sigma(f) \circ \Sigma(f')$:
Obvious since $\Sigma(f) = f$ and $\Sigma(f') = f'$. □

As our first result (Proposition 2.20) states, the two functors are not inverse, but form an adjunction. We shall rely on the following characterisation of adjunctions, based on Definition 9.1 of [2].

Definition 2.19. Let $\mathbf{C}$ and $\mathbf{D}$ be categories, and let $F : \mathbf{D} \to \mathbf{C}$ and $G : \mathbf{C} \to \mathbf{D}$ be functors. Then F is a *left adjoint* of G if one of the following holds:

1. there exists a natural transformation $\eta : I_{\mathbf{D}} \to G \circ F$ (the *unit*) such that for any $c \in \mathbf{C}$, $d \in \mathbf{D}$ and morphism $g : d \to G(c)$ there is a unique morphism $\bar{g} : F(d) \to c$ such that $g = G(\bar{g}) \circ \eta_d$, where η_d is the component of η at d;
2. there exists a natural transformation $\epsilon : F \circ G \to I_{\mathbf{C}}$ (the *counit*) such that for any $c \in \mathbf{C}$, $d \in \mathbf{D}$ and morphism $f : F(d) \to c$ there is a unique morphism $\bar{f} : d \to G(c)$ such that $f = \epsilon_c \circ F(\bar{f})$, where ϵ_c is the component of ϵ at c.

We shall write $F \dashv G$ to denote that F is left adjoint of G and G is right adjoint of F.

Proposition 2.20. *$\Sigma \dashv A$.*

Proof. For any AES $\mathcal{E} = (E, <, \lhd)$, we have $A(\Sigma(\mathcal{E})) = (E, <, \lhd')$ where $\lhd' = < \cup (\lhd \cap \rhd)$.

Let the unit $\eta : \mathcal{E} \to A(\Sigma(\mathcal{E}))$ be defined by $\eta(e) = e$ for all $e \in E$. This is easily seen to be an AES morphism, noting that $\lhd' \subseteq \lhd$ since $\lhd' = < \cup (\lhd \cap \rhd)$ and $< \subseteq \lhd$. Since $\eta(e) = e$, it is obviously a natural transformation. Let $\mathcal{E}$ be an AES and $\mathcal{E}'$ be a PES, and let $g : \mathcal{E} \to A(\mathcal{E}')$ be an AES morphism. Let $\bar{g} : \Sigma(\mathcal{E}) \to \mathcal{E}'$ use the same mapping on events. Then clearly $g(\mathcal{E}) = A(\bar{g}) \circ \eta(\mathcal{E})$ and $\bar{g}$ is the only morphism with this property.

We then show that for any AES morphism $g : \mathcal{E} \to A(\mathcal{E}')$, $\bar{g} : \Sigma(\mathcal{E}) \to \mathcal{E}'$ is a PES morphism, satisfying the conditions of Definition 2.6:

1. This is also a condition of an AES-morphism.
2. If $\bar{g}(e) \#' \bar{g}(e')$ then $g(e) \lhd' g(e')$ and $g(e) \rhd' g(e')$, implying that $e \lhd e'$ and $e \rhd e'$, and therefore $e \# e'$.
3. If $g(e) = g(e')$ and $e \neq e'$ then by definition of an AES morphism, $e \lhd e'$ and $e \rhd e'$, and therefore $e \# e'$. □

Remark 2.21. In [3] it is shown that going from **AES** to **PES** via **Dom** gives the opposite adjunction, that is, $A \dashv D_a \circ P_d$. Thus A has both left and right adjoints.

2.4. General event structures

General event structures, or simply *event structures*, work somewhat differently from PESs or AESs. Instead of causality they have an enabling relation, $\vdash$, so that if a set of events X enables e, $X \vdash e$, then e can happen in the configuration X and any configurations X is a subset of. General event structures also replace conflict with a set of consistent sets of events, so that configurations cannot have any inconsistent finite subsets.

Definition 2.22 *(Event structure [25]).* An *event structure* (ES) is a triple $\mathcal{E} = (E, \mathsf{Con}, \vdash)$, where E is a set of events, $\mathsf{Con} \subseteq_{\text{fin}} 2^E$ is the consistency relation, $\vdash \subseteq \mathsf{Con} \times E$ is the enabling relation, and

1. if $X \in \mathsf{Con}$ and $Y \subseteq X$ then $Y \in \mathsf{Con}$;
2. if $X \vdash e$ and $X \subseteq Y \in \mathsf{Con}$ then $Y \vdash e$.

ES configurations are defined in Definition 2.23. Since conflict has been replaced with consistent sets, we now require all finite subsets of a configuration to be consistent. In order to ensure that the configuration is reachable, we need there to exist a sequence in which the events could have happened.

Definition 2.23 *(Configuration [25]).* Given an ES $\mathcal{E} = (E, \mathsf{Con}, \vdash)$, a *configuration* of $\mathcal{E}$ is a set $X \subseteq E$ such that

1. for all $X' \subseteq_{\text{fin}} X$, $X' \in \mathsf{Con}$;
2. for all $e \in X$, there exists a sequence $e_0, e_1, \ldots, e_n$ such that $e_n = e$ and for all $0 \leq i \leq n$, $\{e_0, \ldots, e_{i-1}\} \vdash e_i$.

Again, the set of all configurations of $\mathcal{E}$ is denoted $\mathsf{Conf}(\mathcal{E})$.

We define an ES morphism in Definition 2.24, giving us the category **ES**.

Definition 2.24 *(ES morphism [25]).* Let $\mathcal{E}_0 = (E_0, \mathsf{Con}_0, \vdash_0)$ and $\mathcal{E}_1 = (E_1, \mathsf{Con}_1, \vdash_1)$ be event structures. An ES morphism is a partial function $f : E_0 \to E_1$ such that

1. for all $e_0 \in E_0$, if $f(e_0) \neq \bot$ and $X \vdash_0 e_0$, then $f(X) \vdash_1 f(e_0)$;
2. if $X \in \mathsf{Con}_0$ then $f(X) \in \mathsf{Con}_1$;
3. for all $e_0, e_0' \in E_0$, if $\{e_0, e_0'\} \in \mathsf{Con}_0$ and $f(e_0) = f(e_0') \neq \bot$ then $e_0 = e_0'$.

Stable event structures, defined in Definition 2.25, are a subset of event structures, which along with ES morphisms between stable event structures make up the category **SES**. An ES is stable if in any given configuration, each event has a unique enabling set.

Definition 2.25 *(Stable event structure [25]).* A *stable* event structure (SES) is an event structure $\mathcal{E} = (E, \mathsf{Con}, \vdash)$ such that if $X \vdash e$, $X' \vdash e$ and $X \cup X' \cup \{e\} \in \mathsf{Con}$ then $X \cap X' \vdash e$.

We then define functors from **PES** to **SES** and from **SES** to **Dom**.

Definition 2.26 *(From **PES** to **SES** [25]).* The functor $P_{ps} : \mathbf{PES} \to \mathbf{SES}$ is defined as:

1. $P_{ps}((E, <, \#)) = (E, \mathsf{Con}, \vdash)$ where $\mathsf{Con} = \{X \subseteq_{\text{fin}} E \mid e, e' \in X \Rightarrow \neg(e \# e')\}$ and $X \vdash e$ if $\{e' \mid e' < e\} \subseteq X \in \mathsf{Con}$;
2. $P_{ps}(f) = f'$ where $f'(X) = \{f(e) \mid e \in X\}$.

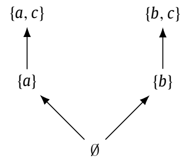


Fig. 5. Domain corresponding to SES $\mathcal{E}$ of Example 2.29.

Remark 2.27. AESs are not as easily mapped into ESs. Consider the AES $\mathcal{E} = (E, <, \lhd)$ where $E = \{a, b\}$ and $a \lhd b$. An attempt to describe $\mathcal{E}$ as a ES $(E, \mathsf{Con}, \vdash)$ where $E = \{a, b\}$ would require $\emptyset \vdash a$ and $\{b\} \in \mathsf{Con}$, but not $\{b\} \vdash a$, which is not possible. We shall see that matters are different in the reversible setting, thanks to SRES enablings having a preventing set.

Definition 2.28 *(From **SES** to **Dom** [25]).* The functor $D_s : \mathbf{SES} \to \mathbf{Dom}$ is defined as:

1. $D_s((E, \mathsf{Con}, \vdash)) = \langle \mathsf{Conf}(\mathcal{E}), \subseteq \rangle$;
2. $D_s(f) = f$.

Example 2.29 shows an SES modelled as a domain.

Example 2.29. $\mathcal{E} = (E, \mathsf{Con}, \vdash)$ where $E = \{a, b, c\}$, $\mathsf{Con} = \{\emptyset, \{a\}, \{b\}, \{a, c\}, \{b, c\}\}$, and $\emptyset \vdash a$, $\emptyset \vdash b$, $\{a\} \vdash c$, and $\{b\} \vdash c$ can be represented by the domain $D_s(\mathcal{E})$ seen in Fig. 5.

We now have all the categories and functors seen in Fig. 2.

3. Configuration systems

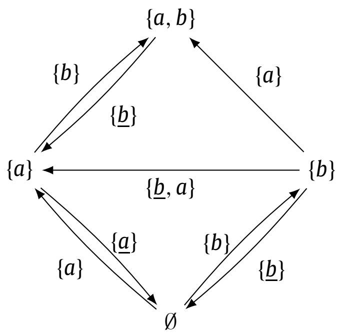
Before we start describing the various categories of reversible event structures, we describe a model of concurrency, into which we can translate the reversible event structures. Obviously domains cannot model reversibility, and so we need a different model. We adopt *configuration systems* [19] (Definition 3.1). These consist of a set of events, E, some of which, F, are reversible, a set C of configurations on these, and an optionally labelled transition relation $\to$ such that if $X \xrightarrow{A \cup \underline{B}} Y$ then the events of A can happen and the events of B can be undone in any order starting from configuration X, resulting in Y. When discussing reversible events we will use $\underline{e}$ to denote reversing e and e^* to denote that e may be performed or reversed and $X + e^*$ to denote $X \cup \{e\}$ if $e^* = e$ and $X \setminus \{e\}$ if $e^* = \underline{e}$. We may also leave out Y when describing such a transition, since it is implied that $Y = (X \setminus B) \cup A$. A CS is shown in Example 3.2.

We define the category of CSs, construct coproducts (Definition 3.5) and products (Definition 3.7), and extend the mappings from AESs and PESs to CSs defined by [19] into functors, along with a new functor from ESs to CSs (Definition 3.14). We also define reachability in CSs and a subcategory of CSs where every configuration is reachable (Definition 3.16) and a functor mapping CSs to their reachable subsystems.

Definition 3.1 *(Configuration system [19]).* A *configuration system* (CS) is a quadruple $\mathcal{C} = (E, F, \mathsf{C}, \to)$ where E is a set of events, $F \subseteq E$ is a set of reversible events, $\mathsf{C} \subseteq 2^E$ is the set of configurations, and $\to \mathsf{C} \times 2^{E \cup \underline{F}} \times \mathsf{C}$ is a labelled transition relation such that if $X \xrightarrow{A \cup \underline{B}} Y$ then:

1. $A \cap X = \emptyset$;
2. $B \subseteq X \cap F$;
3. $Y = (X \setminus B) \cup A$;
4. for all $A' \subseteq A$ and $B' \subseteq B$, we have $X \xrightarrow{A' \cup \underline{B'}} Z \xrightarrow{(A \setminus A') \cup (\underline{B \setminus B'})} Y$ (where $Z = (X \setminus B') \cup A' \in \mathsf{C}$).

Example 3.2 *(Configuration System).* The figure below shows a CS with the events a and b. Notice that there exist reverse transitions, which remove events from the configuration, and that because we have a transition going directly from $\{b\}$ to $\{a\}$, we must be able to go from $\{b\}$ to $\{a\}$ both via $\emptyset$ and via $\{a, b\}$. However, despite there existing paths from $\emptyset$ to $\{a, b\}$ via $\{a\}$ and $\{b\}$, we do not require a direct transition $\emptyset \xrightarrow{\{a,b\}} \{a, b\}$. In future examples we will leave out the labels of the transitions to avoid clutter, as they are easily derived from the configurations.



CS transitions have a lot more freedom than domains in how their transitions allow events to happen, as their transitions are not derived from an ordering. The only requirement of their transitions is that when events can happen concurrently, they must also be able to happen in any order.

We introduce a notion of morphism in Definition 3.3, creating the category **CS**. The two first conditions in this definition ensure that morphisms preserve configurations and transitions, while the third condition corresponds to the third condition in Definitions 2.6, 2.12, and 2.24, ensuring that two events which map to the same event cannot appear in the same configuration.

Definition 3.3 *(CS morphism).* Let $\mathcal{C}_0 = (E_0, F_0, \mathsf{C}_0, \rightarrow_0)$ and $\mathcal{C}_1 = (E_1, F_1, \mathsf{C}_1, \rightarrow_1)$ be configuration systems. A CS morphism is a partial function $f : E_0 \rightarrow E_1$ such that

1. for any $X \in \mathsf{C}_0$, $f(X) \in \mathsf{C}_1$;
2. for any $X, Y \in \mathsf{C}_0$, $A \subseteq E_0$, and $B \subseteq F_0$, if $X \xrightarrow{A \cup \underline{B}}_0 Y$ and $f(A \cup B) \neq \emptyset$ then $f(X) \xrightarrow{f(A) \cup f(\underline{B})}_1 f(Y)$;
3. for all $e_0, e'_0 \in E_0$, if $f(e_0) = f(e'_0) \neq \perp$ and $e_0 \neq e'_0$ then there exists no $X \in \mathsf{C}_0$ such that $e_0, e'_0 \in X$.

Proposition 3.4. ***CS*** *where CSs are objects and CS morphisms are arrows, is a category.*

Proof. Since CS morphisms are partial functions they are obviously associative, and $f(e) = e$ is an identity arrow. Hence we only need to prove composability, meaning that given CSs $\mathcal{C}_0$, $\mathcal{C}_1$, and $\mathcal{C}_2$ and morphisms $f : \mathcal{C}_0 \rightarrow \mathcal{C}_1$ and $g : \mathcal{C}_1 \rightarrow \mathcal{C}_2$, $g \circ f : \mathcal{C}_0 \rightarrow \mathcal{C}_2$ is a morphism. Each condition of Definition 3.3 is proved separately:

1. Since f is a CS morphism, $f(X) \in \mathsf{C}_1$, and since g is a CS morphism, for any $X_1 \in \mathsf{C}_1$, $g(X_1) \in \mathsf{C}_2$, and therefore $g(f(X)) \in \mathsf{C}_2$.
2. Since f is a CS morphism, $f(X) \xrightarrow{f(A) \cup f(\underline{B})}_2 f(Y)$ and, since g is a CS morphism, this implies that $g(f(X)) \xrightarrow{g(f(A)) \cup g(f(\underline{B}))}_2 g(f(Y))$.
3. If $g(f(e_0)) = g(f(e'_0)) \neq \perp$ then $f(e_0) = e_1$ and $f(e'_0) = e'_1$ for some $e_1, e'_1 \in E_1$ such that $g(e_1) = g(e'_1) \neq \perp$. Hence either $e_1 = e'_1$ or no $X_1 \in \mathsf{C}_1$ exists such that $e_1, e'_1 \in X_1$.
   In the first case, since f is a morphism, by definition, if $e_0 \neq e'_0$ then no $X \in \mathsf{C}_0$ exists such that $e_0, e'_0 \in X$.
   In the second, by definition for any $X_0 \in \mathsf{C}_0$, $f(X_0) \in \mathsf{C}_1$, and therefore there cannot exist any $X \in \mathsf{C}_0$ such that $e_0, e'_0 \in X$. □

We construct a coproduct of two CSs in Definition 3.5. A coproduct acts as a ‘choice’ between which of the two CSs to behave like, as illustrated by Fig. 6. We do this by making a CS, which includes both sets of events, with their original configurations and transitions, including the ones going to and from the empty configuration, but not creating any configurations with events from both.

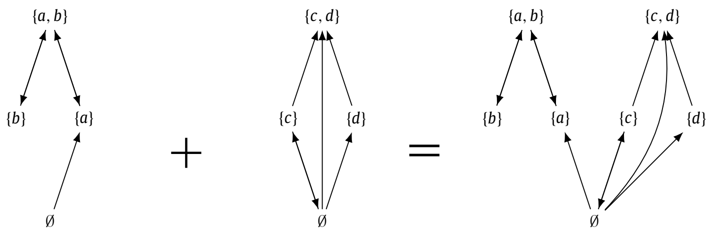


Fig. 6. Example of a coproduct.

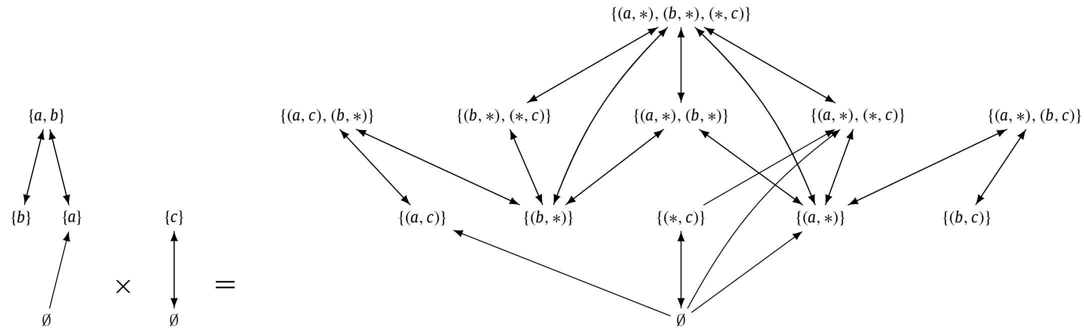


Fig. 7. Example of a Product.

Definition 3.5 *(CS coproduct).* Given CSs $\mathcal{C}_0 = (E_0, F_0, \mathsf{C}_0, \rightarrow_0)$ and $\mathcal{C}_1 = (E_1, F_1, \mathsf{C}_1, \rightarrow_1)$, we can construct $\mathcal{C}_0 + \mathcal{C}_1 = (E, F, \mathsf{C}, \rightarrow)$ with injections i_0 and i_1 where:

1. $E = \{(0, e) \mid e \in E_0\} \cup \{(1, e) \mid e \in E_1\}$;
2. $F = \{(0, e) \mid e \in F_0\} \cup \{(1, e) \mid e \in F_1\}$;
3. for $j \in \{0, 1\}$ and $e \in E_j$ we have $i_j(e) = (j, e)$;
4. $X \in \mathsf{C}$ iff $\exists X_0 \in \mathsf{C}_0.i_0(X_0) = X$ or $\exists X_1 \in \mathsf{C}_1.i_1(X_1) = X$;
5. $X \xrightarrow{A \cup \underline{B}} Y$ iff there exist $j \in \{0, 1\}$, and $X_j, Y_j, A_j, B_j \subseteq E_j$ such that $i_j(X_j) = X$, $i_j(Y_j) = Y$, $i_j(A_j) = A$, $i_j(B_j) = B$, and $X_j \xrightarrow{A_j \cup \underline{B_j}}_j Y_j$.

Proposition 3.6. *Given CSs $\mathcal{C}_0$ and $\mathcal{C}_1$, $\mathcal{C}_0 + \mathcal{C}_1$ is their coproduct in **CS**.*

Proof. Let $\mathcal{C}_0 = (E_0, F_0, \mathsf{C}_0, \rightarrow_0)$, $\mathcal{C}_1 = (E_1, F_1, \mathsf{C}_1, \rightarrow_1)$, and $\mathcal{C}_0 + \mathcal{C}_1 = (E, F, \mathsf{C}, \rightarrow)$. Then, since events from E_i interact with each other in the same way in $\mathcal{C}_0 + \mathcal{C}_1$ as in $\mathcal{C}_i$ and events from E_0 are never in the same configurations as events from E_1, i_0 and i_1 are clearly morphisms.

The coproduct $\mathcal{C}_0 + \mathcal{C}_1$ of $\mathcal{C}_0$ and $\mathcal{C}_1$ must satisfy the universal property that if there exist morphisms $f_0 : \mathcal{C}_0 \rightarrow \mathcal{C}_2$ and $f_1 : \mathcal{C}_1 \rightarrow \mathcal{C}_2$ then there exists a unique morphism $f : \mathcal{C}_0 + \mathcal{C}_1 \rightarrow \mathcal{C}_2$ such that $f \circ i_0 = f_0$ and $f \circ i_1 = f_1$. In this case we define f as $f((j, e)) = f_j(e)$.

Obviously $i_0 \circ f = f_0$ and $i_1 \circ f = f_1$, so we prove f is a morphism, meaning it satisfies the conditions in Definition 3.3:

1. Suppose $X \xrightarrow{A \cup \underline{B}} Y$ in $\mathcal{C}_0 + \mathcal{C}_1$. Then there exists $j \in \{0, 1\}$ and $X_j, Y_j, A_j, B_j \subseteq E_j$ such that $i_j(X_j) = X$, $i_j(Y_j) = Y$, $i_j(A_j) = A$, $i_j(B_j) = B$, and $X_j \xrightarrow{A_j \cup \underline{B_j}}_j Y_j$. Therefore $f_j(X_j) \xrightarrow{f_j(A_j) \cup \underline{f_j(B_j)}}_2 f_j(Y_j)$.
2. Suppose $X \in \mathsf{C}$. Then there exist $j \in \{0, 1\}$ and $X_j \in \mathsf{C}_j$ such that $i_j(X_j) = X$. This implies $f_j(X_j) \in \mathsf{C}_2$.
3. Suppose $f((j, e)) = f((j', e')) \neq \perp$ and $(j, e) \neq (j', e')$. Then either $j = j'$ or $j \neq j'$.
   If $j = j'$ then $f_j(e) = f_j(e')$, implying since $e \neq e'$, there cannot exist $X_j \in \mathsf{C}_j$ such that $e, e' \in X_j$, and therefore there does not exist $X \in \mathsf{C}$ such that $(j, e), (j', e') \in X$.
   If $j \neq j'$ then there does not exist $X \in \mathsf{C}$ such that $(j, e), (j', e') \in X$. $\square$

We also construct a product of CSs in Definition 3.7. A product can be described as the parallel composition of two CSs, as illustrated by Fig. 7. We do this by creating a CS where the events from one set of events can either happen on their own or synchronise with an event from the other CS. When an action happens in one CS without synchronising with an action from the other, we use $*$ instead of the other action. Configurations in this new CS must consist of events such that if one looks only at the events that came from one of the CSs, without considering what they synchronised with, they must be distinct and form a configuration in the original CS. Similarly, projecting a transition onto one of the original CSs, must yield a transition unless all the events being added or removed are events from the other CS that have not synchronised with anything.

Definition 3.7 *(partially synchronous CS product).* Given CSs $\mathcal{C}_0 = (E_0, F_0, \mathsf{C}_0, \rightarrow_0)$ and $\mathcal{C}_1 = (E_1, F_1, \mathsf{C}_1, \rightarrow_1)$, we can construct $\mathcal{C}_0 \times \mathcal{C}_1 = (E, F, \mathsf{C}, \rightarrow)$ with projections π_0 and π_1 where:

1. $E = E_0 \times_* E_1 = \{(e, *) \mid e \in E_0\} \cup \{(*, e) \mid e \in E_1\} \cup \{(e, e') \mid e \in E_0 \text{ and } e' \in E_1\}$;
2. $F = F_0 \times_* F_1 = \{(e, *) \mid e \in F_0\} \cup \{(*, e) \mid e \in F_1\} \cup \{(e, e') \mid e \in F_0 \text{ and } e' \in F_1\}$;
3. for $j \in \{0, 1\}$ and $(e_0, e_1) \in E$ we have $\pi_j((e_0, e_1)) = e_j$.
4. $X \in \mathsf{C}$ if:

   (a) $\pi_0(X) \in \mathsf{C}_0$;
   (b) $\pi_1(X) \in \mathsf{C}_1$;
   (c) for all $e, e' \in X$, if $\pi_0(e) = \pi_0(e') \neq *$ or $\pi_1(e) = \pi_1(e') \neq *$ then $e = e'$.
5. $X \xrightarrow{A \cup \underline{B}} Y$ if:
   (a) if $\pi_0(A \cup B) \neq \emptyset$ then $\pi_0(X) \xrightarrow{\pi_0(A) \cup \underline{\pi_0(B)}}_0 \pi_0(Y)$;
   (b) if $\pi_1(A \cup B) \neq \emptyset$ then $\pi_1(X) \xrightarrow{\pi_1(A) \cup \underline{\pi_1(B)}}_1 \pi_1(Y)$;
   (c) $B \subseteq X$.

Proposition 3.8. *If $\mathcal{C}_0$ and $\mathcal{C}_1$ are CSs then $\mathcal{C}_0 \times \mathcal{C}_1$ is their product in **CS**.*

Proof. Let $\mathcal{C}_0 = (E_0, F_0, \mathsf{C}_0, \rightarrow_0)$, $\mathcal{C}_1 = (E_1, F_1, \mathsf{C}_1, \rightarrow_1)$, and $\mathcal{C}_0 \times \mathcal{C}_1 = (E, F, \mathsf{C}, \rightarrow)$. Then π_0 and π_1 as in Definition 3.7 are clearly morphisms.

A product is formally defined in the following way: if there exist morphisms $f_0 : \mathcal{C}_2 \rightarrow \mathcal{C}_0$ and $f_1 : \mathcal{C}_2 \rightarrow \mathcal{C}_1$ then there exists a unique morphism $f : \mathcal{C}_2 \rightarrow \mathcal{C}_0 \times \mathcal{C}_1$ such that $\pi_0 \circ f = f_0$ and $\pi_1 \circ f = f_1$. We define f as $f(e) = (f_0(e), f_1(e))$, where $f_i(e) = *$ if it is not defined.

Clearly $\pi_0 \circ f = f_0$ and $\pi_1 \circ f = f_1$, so we simply need to show that f is a morphism, meaning it satisfies the conditions of Definition 3.3:

1. If $X \in \mathsf{C}_2$ then $f_0(X) \in \mathsf{C}_0$ and $f_1(X) \in \mathsf{C}_1$, and since f_0 and f_1 are morphisms, there cannot exist $e_2, e_2' \in X$ such that $f_0(e_2) = f_0(e_2') \neq \bot$ or $f_1(e_2) = f_1(e_2') \neq \bot$ and $e_2 \neq e_2'$, implying $f(X) \in \mathsf{C}$.
2. If $X \xrightarrow{A \cup \underline{B}}_2 Y$ then $f_0(X) \xrightarrow{f_0(A) \cup \underline{f_0(B)}}_0 f_0(Y)$ and $f_1(X) \xrightarrow{f_1(A) \cup \underline{f_1(B)}}_1 f_1(Y)$. Therefore $f(X) \xrightarrow{f(A) \cup \underline{f(B)}} f(Y)$.
3. If $f(e) = f(e') \neq \bot$ and $e \neq e'$ then for $j \in \{0, 1\}$, $f_j(e) = f_j(e')$, and there exists $j' \in \{0, 1\}$ such that $f_{j'}(e) = f_{j'}(e') \neq \bot$. This implies there does not exist $X \in \mathsf{C}_2$ such that $e, e' \in X$. □

Functors exist from both **PES**, **AES**, and **ES** to **CS** as described in Definitions 3.9, 3.11, and 3.14. The first two are based on mappings from [19].

The definitions of configurations are not the same here as they were when mapping to domains in the previous section, discarding the conditions which ensured the configurations would be reachable, and keeping only those concerned with conflict. This is because (1) CSs do not have the same requirements of the relationships between their transitions and configurations as domains, and (2) it will become much more difficult to tell which configurations are reachable (in a finite number of steps) once we start discussing reversible event structures. Therefore not restricting ourselves to reachable configurations now will mean that when we map our forward-only event structures into their reversible variants their CS representation will be preserved.

Of the conditions for transitions in Definition 3.9, (a) to (c) are part of Definition 3.1, and (d) simply ensures that e can only happen in transitions going from configurations that contain all of e's causes.

Definition 3.9 *(From **PES** to **CS**).* The functor $C_p : \mathbf{PES} \rightarrow \mathbf{CS}$ is defined as

1. $C_p((E, <, \#)) = (E, \emptyset, \mathsf{C}, \rightarrow)$ where $\mathsf{C} = \{X \subseteq E \mid X \text{ is conflict-free}\}$ and $X \xrightarrow{A} Y$ if:
   (a) $X, Y \in \mathsf{C}$;
   (b) $Y = X \cup A$;
   (c) $A \cap X = \emptyset$;
   (d) for every $e \in A$, $\{e' \in E \mid e' < e\} \subseteq X$;
2. $C_p(f) = f$.

Proposition 3.10. *C_p is a functor from **PES** to **CS**.*

Proof. It was established in [19] that C_p maps PESs to CSs, so we need to show that if $f : \mathcal{E}_0 \rightarrow \mathcal{E}_1$ is a PES morphism then $f : C_p(\mathcal{E}_0) \rightarrow C_p(\mathcal{E}_1)$ is a CS morphism satisfying the conditions of Definition 3.3:

1. Suppose $X \in \mathsf{C}_0$. Then X is conflict-free. Since, by definition of a PES morphism, $f(e_0) \#_1 f(e_0') \Rightarrow e_0 \#_0 e_0'$, this implies that $f(X)$ is conflict-free, and therefore $f(X) \in \mathsf{C}_1$.
2. Suppose $X \xrightarrow{A \cup \underline{B}}_0 Y$. Then $f(X) \xrightarrow{f(A) \cup \underline{f(B)}}_1 f(Y)$ because we can observe that F_0 and therefore B is empty and:
   (a) $f(X), f(Y) \in \mathsf{C}_1$ since $X, Y \in \mathsf{C}_0$, as implied by item 1.
   (b) $f(Y) = f(X) \cup f(A)$ since $Y = X \cup A$.
   (c) $f(A) \cap f(X) = \emptyset$ since $A \cap X = \emptyset$, and by definition of a PES morphism, for all $e_0, e_0' \in E_0$, if $f(e_0) = f(e_0') \neq \bot$ and $e_0 \neq e_0'$ then $e_0 \#_0 e_0'$, implying $e_0, e_0' \notin (X \cup A) = Y \in \mathsf{C}_0$, since Y is conflict-free.

(d) For every $a \in f(A)$, $\{e \in E_1 \mid e <_1 f(a)\} \subseteq f(X)$ because for every $a_0 \in A$, $\{e \in E_0 \mid e <_0 a_0\} \subseteq X$, and by definition of a PES morphism, $\{e_1 \mid e_1 <_1 f(a_0)\} \subseteq \{f(e) \mid e <_0 a_0\}$.

3. Suppose $f(e_0) = f(e'_0) \neq \perp$ and $e_0 \neq e'_0$. By definition of a PES morphism, $e_0 \,\#_0\, e'_0$, and since all $X \in \mathsf{C}_0$ are conflict-free, there exists no $X \in \mathsf{C}_0$ such that $e_0, e'_0 \in X$.

Then to prove it is a functor we simply need to show that:

$C_p(f : \mathcal{E}_1 \to \mathcal{E}_2) = C_p(f) : C_p(\mathcal{E}_1) \to C_p(\mathcal{E}_2)$, which is obvious since $C_p(f) = f$, $C_p(E_0) = E_0$, and $C_p(E_1) = E_1$.

$C_p(1_{\mathcal{E}}) = 1_{C_p(\mathcal{E})}$ since the identity function for all PES and CS objects is $f(e) = e$.

$C_p(f \circ f') = C_p(f) \circ C_p(f')$ since $C_p(f) = f$ and $C_p(f') = f'$. □

In addition to the conditions for transitions in the previous definition, Definition 3.11 also states that if e' prevents e, then e cannot be part of a transition which contains e' either as part of the start configuration or as one of the events being performed simultaneously.

Definition 3.11 *(From **AES** to **CS**).* The functor $C_a : \mathbf{AES} \to \mathbf{CS}$ is defined as:

1. $C_a((E, <, \lhd)) = (E, \emptyset, \mathsf{C}, \to)$ where $\mathsf{C} = \{X \subseteq E \mid \lhd \text{ is well-founded on } X\}$ and $X \xrightarrow{A} Y$ if:
   (a) $X, Y \in \mathsf{C}$;
   (b) $Y = X \cup A$;
   (c) $A \cap X = \emptyset$;
   (d) for every $e \in A$, $\{e' \in E \mid e' < e\} \subseteq X$ and $\{e' \in E \mid e' \rhd a\} \cap (X \cup A) = \emptyset$;
2. $C_a(f) = f$.

Proposition 3.12. *C_a is a functor from **AES** to **CS**.*

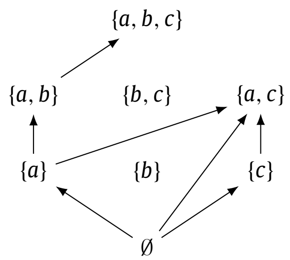
Proof. It was established in [19] that C_a maps AESs to CSs, so we need to show that if $f : \mathcal{E}_0 \to \mathcal{E}_1$ is an AES morphism then $f : C_a(\mathcal{E}_0) \to C_a(\mathcal{E}_1)$ is a CS morphism because it satisfies the conditions of Definition 3.3:

1. Suppose $X \in \mathsf{C}_0$. Then $\lhd_0$ is well-founded on X. Since, by definition of an AES morphism, $f(e_0) \lhd_1 f(e'_0) \Rightarrow e_0 \lhd_0 e'_0$, this implies that $\lhd_1$ is well-founded on $f(X)$, and therefore $f(X) \in \mathsf{C}_1$
2. Suppose $X \xrightarrow{A \cup \underline{B}}_0 Y$. Then $f(X) \xrightarrow{f(A) \cup f(\underline{B})}_1 f(Y)$ because we can observe that F_0 and therefore B must be empty, and:
   (a) $f(X), f(Y) \in \mathsf{C}_1$ because $X, Y \in \mathsf{C}_0$ and we already proved item 1.
   (b) Obviously $f(Y) = f(X) \cup f(A)$, since $Y = X \cup A$.
   (c) $f(A) \cap f(X) = \emptyset$ because $A \cap X = \emptyset$, and by definition of an AES morphism, for all $e_0, e'_0 \in E_0$, if $f(e_0) = f(e'_0) \neq \perp$ and $e_0 \neq e'_0$ then $e_0 \,\#_0\, e'_0$. Hence $e_0, e'_0 \notin (X \cup A) = Y \in \mathsf{C}_0$, since $\lhd_0$ is well-founded on Y.
   (d) For every $a \in f(A)$, there must exist an $a_0 \in A$ such that $f(a_0) = a$, and $\{e \in E_0 \mid e \rhd_0 a_0\} \cap (X \cup A) = \emptyset$. If there exists an $e_1 \in \{e \in E_1 \mid e \rhd_1 a\} \cap (f(X) \cup f(A))$ then there must also exist an $e_0 \in X \cup A$ such that $f(e_0) = e_1$. Since $f(e_0) \rhd_1 f(a_0)$, by definition of an AES morphism, $e_0 \rhd_0 a_0$, but because $\{e \in E_0 \mid e \rhd_0 a_0\} \cap (X \cup A) = \emptyset$ we have a contradiction. Therefore $\{e \in E_1 \mid e \rhd_1 a\} \cap (f(X) \cup f(A)) = \emptyset$.
   (e) For every $a \in f(A)$, $\{e \in E_1 \mid e <_1 a\} \subseteq f(X)$ because for every $a_0 \in A$, $\{e \in E_1 \mid e <_0 a_0\} \subseteq X$, and by definition of an AES morphism, $\{e_1 \mid e_1 <_1 f(a_0)\} \subseteq \{f(e) \mid e <_0 a_0\}$.
3. Suppose $f(e_0) = f(e'_0) \neq \perp$ and $e_0 \neq e'_0$. By definition of an AES morphism, $e_0 \,\#_0\, e'_0$, and since $\lhd_0$ is well-founded on all $X \in \mathsf{C}_0$ we have $e_0, e'_0 \notin X$.

The rest of the proof is identical to that of Proposition 3.10. □

Example 3.13 shows how the AES from Example 2.15 can be mapped to a CS.

Example 3.13. The AES $\mathcal{E} = (E, <, \lhd)$ where $E = \{a, b, c\}$ and $a < b$ and $b \lhd c$, maps to the CS $C_a(\mathcal{E})$, shown in the diagram below:



Notice that, unlike when we used the domain $D_a(\mathcal{E})$ to model $\mathcal{E}$ in Example 2.15, we sometimes get transitions with multiple events, such as $\emptyset \xrightarrow{\{a,c\}} \{a,c\}$, but at other times transitions such as $\emptyset \rightarrow \{a\} \rightarrow \{a,b\}$ exist without $\emptyset \rightarrow \{a,b\}$, allowing us to distinguish between concurrent and sequential events. We also get the unreachable configurations $\{b\}$ and $\{b,c\}$, which were not present in the domain.

The first three conditions on transitions in Definition 3.14 are similar to the two previous definitions, but that condition (d) instead states that e can only be part of a transition if e is associated with an enabling set which is a subset of the starting configuration.

Definition 3.14 *(From **ES** to **CS**).* The functor $C : \mathbf{ES} \rightarrow \mathbf{CS}$ is defined as:

1. $C((E, \mathsf{Con}, \vdash)) = (E, \emptyset, \mathsf{C}, \rightarrow)$ where $X \in \mathsf{C}$ if for all $X' \subseteq_{\text{fin}} X$, $X' \in \mathsf{Con}$, and $X \xrightarrow{A} Y$ if:
   (a) $X, Y \in \mathsf{C}$;
   (b) $Y = X \cup A$;
   (c) $A \cap X = \emptyset$;
   (d) for every $e \in A$, there exists an $X' \subseteq_{\text{fin}} X$ such that $X' \vdash e$;
2. $C(f) = f$.

Proposition 3.15. *C is a functor from **ES** to **CS**.*

Proof. We first prove that for any ES $\mathcal{E} = (E, \mathsf{Con}, \vdash)$, $C(\mathcal{E}) = (E, F, \mathsf{C}, \rightarrow)$ is a CS, meaning that whenever $X \xrightarrow{A \cup \underline{B}} Y$:

1. $A \cap X = \emptyset$, $B \subseteq F = \emptyset$, and $Y = X \cup A$ according to the definition of C.
2. For all $A' \subseteq A$, we have $X \xrightarrow{A'} X \cup A' \xrightarrow{A \setminus A'} Y$ because the conditions for transitions in Definition 3.14 are clearly satisfied for both transitions.

We then show that if $f : \mathcal{E}_0 \rightarrow \mathcal{E}_1$ is an ES morphism then $f : C(\mathcal{E}_0) \rightarrow C(\mathcal{E}_1)$ is a CS morphism satisfying the conditions of Definition 3.3:

1. Suppose $X \in \mathsf{C}_0$. Then for all $X' \subseteq_{\text{fin}} X$, $X' \in \mathsf{Con}_0$. Hence $f(X') \in \mathsf{Con}_1$, and therefore $f(X) \in \mathsf{C}_1$.
2. Suppose $X \xrightarrow{A \cup \underline{B}}_0 Y$. Then $f(X) \xrightarrow{f(A) \cup f(\underline{B})}_1 f(Y)$ because we can observe that F_0 and therefore B must be empty, and:
   (a) $f(X), f(Y) \in \mathsf{C}_1$ since $X, Y \in \mathsf{C}_0$, this is implied by the first item.
   (b) Obviously $f(Y) = f(X) \cup f(A)$, since $Y = X \cup A$.
   (c) $f(A) \cap f(X) = \emptyset$ because $A \cap X = \emptyset$, and for all $e, e' \in Y$, $\{e, e'\} \in \mathsf{Con}_0$, meaning either $e = e'$ or $f(e) \neq f(e')$.
   (d) For every $a \in f(A)$, there exists an $X' \subseteq_{\text{fin}} f(X)$ such that $X' \vdash_1 a$ because for every $a_0 \in A$, there exists an $X'_0 \subseteq_{\text{fin}} X$ such that $X'_0 \vdash_0 a_0$, implying $f(X'_0) \vdash_1 f(a_0)$.
3. Suppose $f(e_0) = f(e'_0) \neq \perp$ and $e_0 \neq e'_0$. Then $\{e_0, e'_0\} \notin \mathsf{Con}_0$, and therefore there does not exist $X \subseteq E_0$ such that $e_0, e'_0 \in X$ and for all $X' \subseteq_{\text{fin}} X$, $X' \in \mathsf{Con}_0$.

Having proved that C maps from **ES** to **CS** we prove it to be a functor similarly to previous functors. □

We will usually not be interested in any part of a CS which is not reachable from the empty configuration $\emptyset$, and so we will define a functor to remove those parts. However, we first define reachability in Definition 3.16.

Definition 3.16 *(Reachable CS).* Given a CS $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$, a configuration $X \in \mathsf{C}$ is *reachable* if there exists some sequence $\emptyset \xrightarrow{A_0 \cup \underline{B_0}} \dots \xrightarrow{A_n \cup \underline{B_n}} X$. A configuration $X \in \mathsf{C}$ is *forwards reachable* if there exists some sequence $\emptyset \xrightarrow{A_0} \dots \xrightarrow{A_n} X$. We say that $\mathcal{C}$ is reachable if all configurations in C are reachable. We let **RCS** be the category consisting of reachable CSs and the CS morphisms between them.

We then define the reachable subsystem of a CS and a functor Reach restricting the CS to said subsystem in Definition 3.17.

Definition 3.17 *(Restricting to Reachable).* Given a CS $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$, the *reachable part* of $\mathcal{C}$, $(E, F, \mathsf{C}', \rightarrow')$, is defined as:

1. $\mathsf{C}' = \{X \mid X \in \mathsf{C},\ X \text{ is reachable}\}$
2. $X \xrightarrow{A \cup \underline{B}}{}' X'$ if $X \xrightarrow{A \cup \underline{B}} X'$ and $X, X' \in \mathsf{C}'$

We define a functor $\mathsf{Reach} : \mathbf{CS} \rightarrow \mathbf{RCS}$ such that:

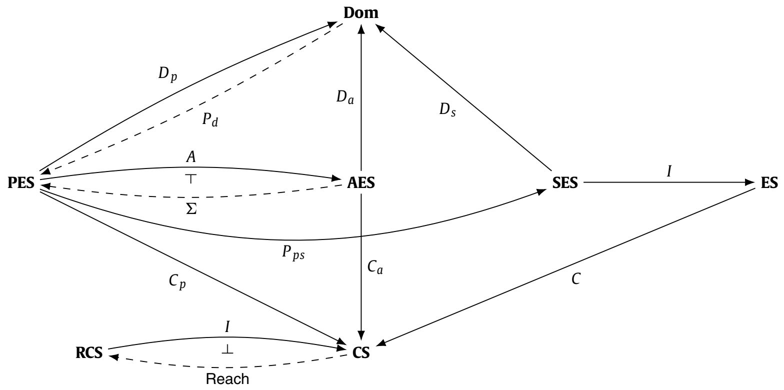


Fig. 8. Categories of forward-only event structures and CSs. We use dashed lines to indicate functors which do not commute with the rest of the diagram.

1. $\mathsf{Reach}(\mathcal{C})$ is the reachable part of $\mathcal{C}$
2. $\mathsf{Reach}(f) = f$

Proposition 3.18. Reach *is a functor from* ***CS*** *to* ***RCS****.*

Proof. First we show that Reach is a mapping, meaning:

1. If $\mathcal{C}$ is a CS then $\mathsf{Reach}(\mathcal{C})$ is an RCS.
2. If f is a CS morphism then $\mathsf{Reach}(f)$ is an RCS morphism.

This holds because:

1. Obvious.
2. Here we need to show that for any CSs $\mathcal{C}_0 = (E_0, F_0, \mathsf{C}_0, \rightarrow_0)$ and $\mathcal{C}_1 = (E_1, F_1, \mathsf{C}_1, \rightarrow_1)$ and CS morphism $f : \mathcal{C}_0 \rightarrow \mathcal{C}_1$, if $X \in \mathsf{C}_0$ is reachable then $f(X)$ is reachable in C_1. But clearly, if $\emptyset \xrightarrow{A_0 \cup \underline{B_0}} \dots \xrightarrow{A_n \cup \underline{B_n}} X$, then $\emptyset \xrightarrow{f(A_0) \cup \underline{f(B_0)}} \dots \xrightarrow{f(A_n) \cup \underline{f(B_n)}} f(X)$.
   In addition, since all reachable configurations were configurations of the original CS, condition 3 of Definition 3.3 is clearly satisfied.

The rest of the proof is similar to previous functors. □

Proposition 3.19. *Let I be the inclusion functor I :* ***RCS*** *→* ***CS****. Then $I \dashv$* Reach*.*

Proof. For any CS $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$ we define counit $\epsilon : I(\mathsf{Reach}(\mathcal{C})) \rightarrow \mathcal{C}$ such that for all $e \in E$, $\epsilon(e) = e$. Clearly ϵ is a CS morphism.

We then show that given an RCS $\mathcal{C}$, a CS $\mathcal{C}'$, and a morphism $f : I(\mathcal{C}) \rightarrow \mathcal{C}'$, $f : \mathcal{C} \rightarrow \mathsf{Reach}(\mathcal{C}')$ is an RCS morphism. As established in the proof of Proposition 3.18, if X is a reachable configuration, then $f(X)$ is a reachable configuration, and the result follows. □

RCS is a full subcategory of **CS**, i.e. includes all the functors between reachable CSs. Moreover, Proposition 3.19 establishes that **RCS** is coreflective in **CS**, meaning its inclusion functor has a right adjoint.

We then have the categories seen in Fig. 8, which illustrates that in addition to modelling reversibility, CSs are more powerful than domains even in the forward-only world, as they are able to model non-stable ESs.

4. Reversible prime event structures

We are now ready to start adding reversibility to our event structures, with the goal of defining the categories and functors seen in the bottom half of Fig. 9.

We begin with PESs. *Reversible prime event structures* (Definition 4.1) consist of events, some of which may be reversible, causality and conflict similar to a PES (though conflict is not necessarily hereditary), reverse causality, which works similarly to causality, in that $e \prec \underline{e'}$ means e' can only be reversed in configurations containing e, and finally prevention, which resembles the asymmetric conflict of AESs, in that $e \rhd \underline{e'}$ means that e' can only be reversed in configurations not containing e. Example 4.2 shows how we can use these to describe a reversible CCS process with causal reversibility. We define the

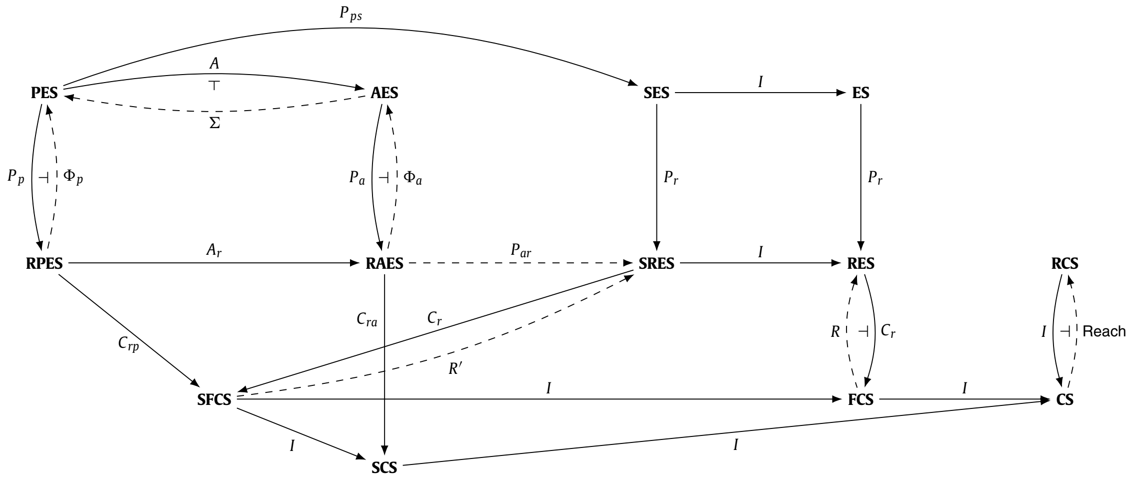
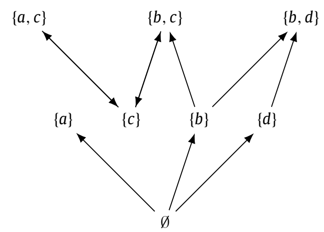


Fig. 9. Categories of event structures and functors between them: We extend Fig. 2 by categorically defining RPESs, RAESs, CSs, P_p, Φ_p, P_a, Φ_a, C_p, C_{pr}, C_a, C_{ra}, and A_r [19] and RESs and P_r [21]. The categories **SRES** (Definition 7.1), **SCS**, **SFCS** (Definition 7.6), **RCS**, and **FCS** (Section 6.11), and functors P_{pr}, P_{ar}, Reach, C_r, C, and R are new, as well as the noted adjunctions. We use dashed lines to indicate functors which do not commute with the rest of the diagram.

$\mathcal{E} = (E, F, <, \#, \prec, \rhd)$ where:
$E = \{a, b, c, d\}$
$F = \{a, b\}$
$b < c$
$a \# b$, $a \# d$, $c \# d$
$c \prec \underline{a}$, $c \prec \underline{b}$, $a \prec \underline{a}$, $b \prec \_$
$d \rhd \underline{b}$

(a) RPES $\mathcal{E}$



(b) The CS derived from $\mathcal{E}$

Fig. 10. Example of an RPES and corresponding CS.

category of RPESs, along with coproducts (Definition 4.6), and extend the mappings from PESs to RPESs, and from RPESs to PESs and CSs defined by [19] into functors. We show that the functors between RPESs and PESs form an adjunction (Proposition 4.16).

Definition 4.1 *(RPES [19]).* A *reversible prime event structure* (RPES) is a sextuple $\mathcal{E} = (E, F, <, \#, \prec, \rhd)$ where E is the set of events and

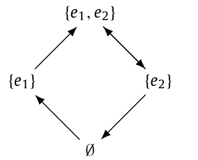
1. $<$ is an irreflexive partial order such that for every $e \in E$, $\{e' \in E \mid e' < e\}$ is finite and conflict-free;
2. $\#$ is irreflexive and symmetric;
3. if $e < e'$ then not $e \# e'$;
4. $F \subseteq E$ is the set of reversible events;
5. $\rhd \subseteq E \times \underline{F}$ is the prevention relation;
6. $\prec \subseteq E \times \underline{F}$ is the reverse causality relation where for each $e \in F$, $e \prec \underline{e}$ and $\{e' \mid e' \prec \underline{e}\}$ is finite and conflict-free;
7. if $e \prec \underline{e'}$ then not $e \rhd \underline{e'}$;
8. $\#$ is hereditary with respect to *sustained causation* $\prec\!\!\prec$, where $e \prec\!\!\prec e'$ means that $e < e'$ and if $e \in F$ then $e' \rhd \underline{e}$;
9. $\prec\!\!\prec$ is transitive.

Example 4.2. Recall Example 1.1 wherein we wanted to model the reversible CCS process $a \mid \overline{a}.b$. This can be modelled by an RPES wherein we have two b events b_τ and $b_{\overline{a}}$ such that $\tau < b_\tau$ and $\overline{a} < b_{\overline{a}}$, $b_\tau \# b_{\overline{a}}$, $a \# \tau$, and $\overline{a} \# \tau$, and $b_\tau \rhd \underline{\tau}$ and $b_{\overline{a}} \rhd \underline{\overline{a}}$.

As previously, in order to define the category **RPES**, we need a notion of morphism, which we describe in Definition 4.3. An RPES morphism can be seen as a combination of a PES morphism for the forwards part and an AES morphism for the reverse part.

$\mathcal{E}' = (E', F', <', \#', \prec', \rhd')$ where:
$E' = \{e_1, e_2\}$
$F' = \{e_1, e_2\}$
$e_1 <' e_2$
$e_2 \prec' \underline{e_1}$, $e_1 \prec' \underline{e_1}$, $e_2 \prec' \underline{e_2}$
$e_1 \rhd' \underline{e_2}$

(a) RPES $\mathcal{E}'$



(b) The CS derived from $\mathcal{E}'$

Fig. 11. Another RPES and corresponding CS.

Definition 4.3 *(RPES morphism).* Let $\mathcal{E}_0 = (E_0, F_0, <_0, \#_0, \prec_0, \rhd_0)$ and $\mathcal{E}_1 = (E_1, F_1, <_1, \#_1, \prec_1, \rhd_1)$ be RPESs. An RPES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$ is a partial function $f : E_0 \to E_1$ such that

1. for all $e \in E_0$, if $f(e) \neq \bot$ then $\{e_1 \mid e_1 <_1 f(e)\} \subseteq \{f(e') \mid e' <_0 e\}$;
2. for all $e, e' \in E_0$, if $f(e) \neq \bot \neq f(e')$ and $f(e) \#_1 f(e')$ then $e \#_0 e'$;
3. for all $e \in F_0$, if $f(e) \neq \bot$ then $\{e_1 \mid e_1 \prec_1 \underline{f(e)}\} \subseteq \{f(e') \mid e' \prec_0 \underline{e}\}$;
4. for all $e \in E_0$ and $e' \in F_0$, if $f(e) \neq \bot \neq f(e')$ and $f(e) \rhd_1 \underline{f(e')}$ then $e \rhd_0 \underline{e'}$;
5. for all $e, e' \in E_0$, if $f(e) = f(e') \neq \bot$ and $e \neq e'$ then $e \#_0 e'$;
6. $f(F_0) \subseteq F_1$.

Proposition 4.4. ***RPES****, where RPESs are objects and RPES morphisms are arrows, is a category.*

Proof. Since RPES morphisms are partial functions they are obviously associative, and $f(e) = e$ is an identity arrow; so we only need to prove composability, meaning that given RPESs $\mathcal{E}_0$, $\mathcal{E}_1$, and $\mathcal{E}_2$ and morphisms $f : \mathcal{E}_0 \to \mathcal{E}_1$ and $g : \mathcal{E}_1 \to \mathcal{E}_2$, $g \circ f : \mathcal{E}_0 \to \mathcal{E}_2$ is a morphism.

Each of the six conditions of Definition 4.3 is proved separately:

1. If $g(f(e)) \neq \bot$ then there must exist $e_1 \in E_1$ and $e_2 \in E_2$ such that $f(e) = e_1$ and $g(e_1) = e_2$. Since f and g are morphisms, $\{e'_1 \mid e'_1 <_1 e_1\} \subseteq \{f(e') \mid e' <_0 e\}$ and $\{e'_2 \mid e'_2 <_2 e_2\} \subseteq \{g(e'_1) \mid e'_1 <_1 e_1\}$, and therefore $\{e'_2 \mid e'_2 <_2 g(f(e))\} \subseteq \{g(f(e')) \mid e' <_0 e\}$.
2. If $g(f(e)) \neq \bot \neq g(f(e'))$ then there must exist $e_1, e'_1 \in E_1$ and $e_2, e'_2 \in E_2$ such that $f(e) = e_1$, $f(e') = e'_1$, $g(e_1) = e_2$, and $g(e'_1) = e'_2$. By definition of a morphism, if $e_2 \# e'_2$ then $e_1 \# e'_1$, and if $e_1 \# e'_1$ then $e \# e'$.
3. Similar to the first condition.
4. Similar to the second condition.
5. If $g(f(e)) = g(f(e')) \neq \bot$ then there must exist $e_1, e'_1 \in E_1$ and $e_2 \in E_2$ such that $f(e) = e_1$, $f(e') = e_1$, and $g(e_1) = g(e'_1) = e_2$. If $e_1 = e'_1$ then, since f is a morphism, $e \#_0 e'$. If $e_1 \neq e'_1$ then $e_1 \#_1 e'_1$, implying that $e \#_0 e'$.
6. Obvious, since $f(F_0) \subseteq F_1$ and $g(F_1) \subseteq F_2$. □

Example 4.5 *(RPES morphism).* Consider the RPESs $\mathcal{E}$ from Fig. 10 and $\mathcal{E}'$ from Fig. 11. Then

$$f(e) = \begin{cases} e_1 & \text{if } e \in \{a, b\} \\ e_2 & \text{if } e = c \end{cases}$$

is an RPES morphism between them, and a CS morphism between their associated CSs.

We also construct a coproduct of two RPESs in Definition 4.6. A coproduct of two RPESs, which can be represented by the CSs from Fig. 6, is shown in Example 4.8. Similarly to Definition 3.5, we keep the two sets of events separate, and with the same causality, conflict, reverse causality, and prevention as they had in their respective original RPESs, but add conflict and prevention between the events that come from different RPESs.

Definition 4.6 *(RPES coproduct).* Given RPESs $\mathcal{E}_0 = (E_0, F_0, <_0, \#_0, \prec_0, \rhd_0)$ and $\mathcal{E}_1 = (E_1, F_0, <_1, \#_1, \prec_1, \rhd_1)$, we can construct $\mathcal{E}_0 + \mathcal{E}_1 = (E, F, <, \#, \prec, \rhd)$ with injections i_0 and i_1 where:

1. $E = \{(0, e) \mid e \in E_0\} \cup \{(1, e) \mid e \in E_1\}$;
2. $F = \{(0, e) \mid e \in F_0\} \cup \{(1, e) \mid e \in F_1\}$;
3. $(j, e) < (j', e')$ iff $j = j'$ and $e <_j e'$;
4. $(j, e) \# (j', e')$ iff $j \neq j'$ or $e \#_j e'$;
5. $(j, e) \prec \underline{(j', e')}$ iff $j = j'$ and $e \prec_j \underline{e'}$;
6. $(j, e) \rhd \underline{(j', e')}$ iff $j \neq j'$ or $e \rhd_j \underline{e'}$;
7. for $e \in E_j$, $i_j(e) = (j, e)$ for $j \in \{0, 1\}$.

Proposition 4.7. *Given RPESs $\mathcal{E}_0$ and $\mathcal{E}_1$, $\mathcal{E}_0 + \mathcal{E}_1$ is their coproduct in **RPES**.*

Proof. Let $\mathcal{E}_0 = (E_0, F_0, <_0, \#_0, \prec_0, \triangleright_0)$, $\mathcal{E}_1 = (E_1, F_0, <_1, \#_1, \prec_1, \triangleright_1)$, and $\mathcal{E}_0 + \mathcal{E}_1 = \mathcal{E} = (E, F, <, \#, \prec, \triangleright)$. Then it is straightforward to show that $\mathcal{E}$ is an RPES, and i_0 and i_1 are morphisms, so we simply need to prove that if there exists an RPES $\mathcal{E}_2 = (E_2, F_2, <_2, \#_2, \prec_2, \triangleright_2)$ and morphisms $f_0 : \mathcal{E}_0 \to \mathcal{E}_2$ and $f_1 : \mathcal{E}_1 \to \mathcal{E}_2$, then there exists a unique RPES morphism $f : \mathcal{E} \to \mathcal{E}_2$ such that $f \circ i_0 = f_0$ and $f \circ i_1 = f_1$.

Since $E_0 + E_1$, i_0, and i_1 comprise a coproduct in the category of sets and partial functions, f must be unique.

We define f as $f((j, e)) = f_j(e)$, giving us $i_0 \circ f = f_0$ and $i_1 \circ f = f_1$, and prove f to be a morphism:

1. Suppose $e_2 <_2 f(e)$. Then there exists a $j \in \{0, 1\}$ and an $e_j \in E_j$ such that $e = (j, e_j)$ and $e_2 <_2 f_j(e_j)$. Since f_j is a morphism, there must exist an $e'_j \in E_j$ such that $f_j(e'_j) = e_2$ and $e'_j <_j e_j$. Hence that $(j, e'_j) < e$.
2. Suppose $f(e') \#_2 f(e)$. The $e' = (j', e_{j'})$ and $e = (j, e_j)$ for some $j, j' \in \{0, 1\}$, $e_{j'} \in E_{j'}$, and $e_j \in E_j$, and therefore $f(e) = f_j(e_j)$ and $f(e') = f_{j'}(e_{j'})$.
   If $j = j'$, then $e_{j'} \#_j e_j$, and therefore $e_{j'} \#_j e_j$, $(j, e_{j'}) \# (j, e_j)$.
   If $j \neq j'$ then $e' \# e$.
3. Suppose $e_2 \prec_2 \underline{f(e)}$. Then there exists a $j \in \{0, 1\}$ and an $e_j \in E_j$ such that $e = (j, e_j)$ and $e_2 \prec_2 \underline{f_j(e_j)}$. Since f_j is a morphism, there must exist an $e'_j \in E_j$ such that $f_j(e'_j) = e_2$ and $e'_j \prec_j \underline{e_j}$. Therefore $(j, e'_j) \prec \underline{e}$, and obviously $f(j, e'_j) = e_2$.
4. Suppose $f(e') \triangleright_2 \underline{f(e)}$. Then there must exist a $j \in \{0, 1\}$ and $e_j, e'_j \in E_j$ such that $f_j(e_j) = f(e)$, $f_j(e'_j) = f(e')$ and $e'_j \#_j e_j$. Since $e'_j \triangleright_j \underline{e_j}$, $(j, e'_j) \triangleright \underline{(j, e_j)}$.
5. Suppose $f(e) = f(e') \neq \bot$. Then $e = (j, e_j)$ and $e' = (j', e_{j'})$ and $f_j(e_j) = f(e) = f(e') = f_{j'}(e_{j'})$.
   If $j = j'$ then $e \neq e'$ implies that $e_j \neq e_{j'}$. Hence $e_j \#_j e_{j'}$ and therefore $e \# e'$.
   If $j \neq j'$ then obviously $e \# e'$.
6. Suppose $e \in F$ and $f(e) \neq \bot$. Then there exist $j \in \{0, 1\}$ and $e_j \in F_j$ such that $e = (j, e_j)$ and $f_j(e_j) = f(e)$. Since $e_j \in F_j$, $f_j(e) \in F_2$. □

Example 4.8 *(RPES coproduct).* Given RPESs corresponding to the CSs in Fig. 6, $\mathcal{E}_0 = (E_0, F_0, <_0, \#_0, \prec_0, \triangleright_0)$ and $\mathcal{E}_1 = (E_1, F_1, <_1, \#_1, \prec_1, \triangleright_1)$ where $E_0 = \{a, b\}$, $F_0 = \{a, b\}$, $a <_0 b$, $a \prec_0 \underline{b}$ and $E_1 = \{c, d\}$, $F_1 = \{c\}$, and $d \triangleright_1 \underline{c}$, the coproduct $\mathcal{E}_0 + \mathcal{E}_1$ is $(E, F, <, \#, \prec, \triangleright)$, where

$E = \{(0, a), (0, b), (1, c), (1, d)\}$,
$F = \{(0, a), (0, b), (1, c)\}$,
$(0, a) < (0, b)$,
$(0, a) \prec \underline{(0, b)}$,
$(0, a) \# (1, c)$, $(0, a) \# (1, d)$, $(0, b) \# (1, c)$, $(0, b) \# (1, d)$,
$(0, a) \triangleright \underline{(1, c)}$, $(0, b) \triangleright \underline{(1, c)}$, $(1, c) \triangleright \underline{(0, a)}$, $(1, d) \triangleright \underline{(0, a)}$, $(1, c) \triangleright \underline{(0, b)}$, and $(1, d) \triangleright \underline{(0, b)}$
$(1, d) \triangleright \underline{(1, c)}$.

We do not give a way to construct a product of RPESs, as we have not found a notion of PES product which translates to a reversible setting. One way to construct a PES product can be seen in [23], where the definition is similar to mapping both event structures to domains, and mapping the product of those domains to a prime event structure. Since we do not have mappings from CSs to RPESs or RAESs, this is not a solution we can use.

Another way of looking at the PES product is that we can split an event, similar to in Example 4.2, based on which of the events of the other PES the event's causes synchronise with. However, splitting an event to allow it to have multiple causes does not work when dealing with reverse causation, as shown by Example 4.9. We shall see later in Remark 4.22 that splitting is not an issue in all RPESs, and so we can construct products in a subcategory.

Example 4.9. Consider an RPES where we want $\underline{a}$ to be caused by either b or c. We attempt to accomplish this by splitting a into a_b and a_c such that $b \prec \underline{a_b}$ and $c \prec \underline{a_c}$. However, we would like to have a computation $\emptyset \xrightarrow{a} \{a\}$ after which we have both $\{a\} \xrightarrow{b} \{a, b\} \xrightarrow{\underline{a}} \{b\}$ and $\{a\} \xrightarrow{c} \{a, c\} \xrightarrow{\underline{a}} \{c\}$ are possible computations. We do not have this. By contrast, when splitting because of forwards computation, we know when we first perform the split event which of its causes is being used, and so we do not have this issue.

We leave it as an open question whether products exist in general in the category **RPES**.

In order to use CSs as a model for RPESs, [19] defined a mapping, which we extend to the functor C_{rp}, from RPESs to CSs, illustrated in Figs. 10 and 11. It uses conflict-free sets of events as configurations, and the transition relation is similar to the one generated by PESs and AESs, except that we add reverse transitions, which remove events instead of adding them, and treat reverse causality and prevention similarly to forward causality and asymmetric conflict.

Definition 4.10 *(From **RPES** to **CS**).* The functor C_{rp} : **RPES** → **CS** is defined as

1. $C_{rp}((E, F, <, \#, \prec, \triangleright)) = (E, F, \mathsf{C}, \rightarrow)$ where $\mathsf{C} = \{X \subseteq E \mid X \text{ is conflict-free}\}$, and $X \xrightarrow{A \cup \underline{B}} Y$ if:
   (a) $X, Y \in \mathsf{C}$, $A \subseteq E$, and $B \subseteq F$;
   (b) $Y = (X \setminus B) \cup A$;
   (c) $A \cap X = \emptyset$;
   (d) $B \subseteq X$;
   (e) $X \cup A$ is conflict-free;
   (f) for every $e \in A$, if $e' < e$ then $e' \in X \setminus B$;
   (g) for every $e \in B$, if $e' \prec \underline{e}$ then $e' \in X \setminus (B \setminus \{e\})$;
   (h) for every $e \in B$, if $e' \triangleright \underline{e}$ then $e' \notin X \cup A$.
2. $C_{rp}(f) = f$.

Proposition 4.11. *C_{rp} is a functor from **RPES** to **CS**.*

Proof. [19] showed that given an RPES $\mathcal{E}$, $C_{rp}(\mathcal{E})$ is a CS, so we show that given an RPES morphism $f : \mathcal{E}_0 \rightarrow \mathcal{E}_1$, $f : C_{rp}(\mathcal{E}_0) \rightarrow C_{rp}(\mathcal{E}_1)$ is a CS morphism, satisfying the conditions of Definition 3.3

1. Suppose $X \in \mathsf{C}_0$. Then X is conflict-free. Since, by definition of an RPES morphism, $f(e_0) \#_1 f(e'_0) \Rightarrow e_0 \#_0 e'_0$, this implies $f(X)$ is conflict-free, and therefore $f(X) \in \mathsf{C}_1$.
2. Suppose $X \xrightarrow{A \cup \underline{B}} Y$. Then $f(X) \xrightarrow{f(A) \cup \underline{f(B)}}_1 f(Y)$ according to Definition 4.10 because:
   (a) Since $X, Y \in \mathsf{C}_0$, $f(X), f(Y) \in \mathsf{C}_1$ according to previous item.
   (b) Obviously $f(Y) = (f(X) \setminus f(B)) \cup f(A)$, since $Y = (X \setminus B) \cup A$ and $B \subseteq X$.
   (c) $f(A) \cap f(X) = \emptyset$ because $A \cap X = \emptyset$, and by definition of an RPES morphism, for all $e_0, e'_0 \in E_0$, if $f(e_0) = f(e'_0) \neq \perp$ and $e_0 \neq e'_0$ then $e_0 \# e'_0$, implying $e_0, e'_0 \notin (X \cup A)$, since $X \cup A$ is conflict-free.
   (d) $f(B) \subseteq f(X)$ because $B \subseteq X$.
   (e) $f(X) \cup f(A)$ is conflict-free because $X \cup A$ is conflict-free, and for all $e_0, e'_0 \in E_0$, if $f(e_0) \#_1 f(e'_0)$ then $e_0 \#_0 e'_0$.
   (f) For every $a_1 \in f(A)$, if $c_1 <_1 a_1$ then $c_1 \in f(X) \setminus f(B)$ because for every $a_0 \in A$, if $c_0 <_0 a_0$ then $c_0 \in X \setminus B$, implying $f(c) \in f(X) \setminus f(B)$, and by definition of an RPES $\{e_1 \mid e_1 <_1 f(a_0)\} \subseteq \{f(e') \mid e' <_0 a_0\}$.
   (g) For every $b_1 \in f(B)$, if $d_1 \prec_1 \underline{b_1}$ then $d_1 \in f(X) \setminus (f(B) \setminus \{b_1\}$ for similar reasons to above.
   (h) For every $b_1 \in f(B)$, if $d_1 \triangleright_1 \underline{b_1}$ then $d_1 \notin f(X) \cup f(A)$ because, since $b_1 \in f(B)$ there must exist a $b_0 \in B$ such that $f(b_0) = b_1$. If $d_1 \in f(X) \cup f(A)$ then there must exist a $d_0 \in X \cup A$ such that $f(d_0) = d_1$. Since $f(d_0) \triangleright_1 f(b_0)$, by definition of an RAES morphism $d_0 \triangleright_0 b_0$, implying $d_0 \notin X \cup A$ which is a contradiction.
3. Suppose $f(e_0) = f(e'_0) \neq \perp$ and $e_0 \neq e'_0$. By definition of an RPES morphism, $e_0 \#_0 e'_0$, and since all $X \in \mathsf{C}_0$ are conflict-free, there cannot exist $X \in \mathsf{C}$ such that $e_0, e'_0 \in X$.

Having shown that C_{rp} maps from **RPES** to **CS**, the rest of the proof is similar to previous functors. □

We also have functors between PES and RPES in Definitions 4.12 and 4.13, likewise based on mappings from [19]. Since # is only closed under conflict heredity with respect to sustained causation in an RPES, turning an RPES into a PES requires expanding conflict to be closed under all causality.

Definition 4.12 *(From **PES** to **RPES**).* The functor P_p : **PES** → **RPES** is defined as

1. $P_p((E, <, \#)) = (E, \emptyset, <, \#, \emptyset, \emptyset)$;
2. $P_p(f) = f$.

Definition 4.13 *(From **RPES** to **PES**).* The functor Φ_p : **RPES** → **PES** is defined as

1. $\Phi_p((E, F, <, \#, \prec, \triangleright)) = (E, <, \#')$ where $\#'$ is # closed under conflict heredity and symmetry using the rules:

$$\frac{a \# b}{a \#' b} \quad \frac{a \#' b}{b \#' a} \quad \frac{a \#' b < c}{a \#' c}$$

2. $\Phi_p(f) = f$.

For an example of an RPES being mapped to a PES, see Example 4.17.

Proposition 4.14. *P_p is a functor from **PES** to **RPES**.*

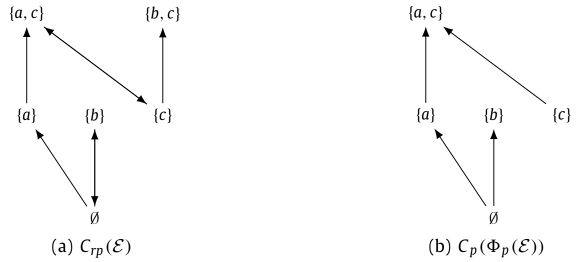


Fig. 12. The CSs corresponding to the RPES and PES of Example 4.17.

Proof. [19] showed that given a PES $\mathcal{E}$, $P_p(\mathcal{E})$ is an RPES; so it remains to show that given a PES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$, $f : P_p(\mathcal{E}_0) \to P_p(\mathcal{E}_1)$ is an RPES morphism. This is because $F_0 = F_1 = \emptyset$, and items 1, 2 and 6 of Definition 4.3 are implied by the definition of a PES morphism.

Having shown that P_p maps from **PES** to **RPES**, the rest of the proof is similar to previous functors. □

Proposition 4.15. Φ_p *is a functor from* ***RPES*** *to* ***PES****.*

Proof. [19] showed that given an RPES $\mathcal{E}$, $\Phi_p(\mathcal{E})$ is a PES, so we show that given an RPES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$, $f : \Phi_p(\mathcal{E}_0) \to \Phi_p(\mathcal{E}_1)$ is a PES morphism, satisfying the conditions of Definition 2.6

1. This condition is the same as condition 1 of Definition 4.3 of an RPES morphism.
2. Suppose $f(e_0) \#'_1 f(e'_0)$. Then either $f(e_0) \#_1 f(e'_0)$, $f(e'_0) \#'_1 f(e_0)$, or there exists some $e_1 \in E_1$ such that $f(e''_0) < f(e'_0)$ and $f(e_0) \#'_1 f(e''_0)$.
If $f(e_0) \#_1 f(e'_0)$ then obviously $e_0 \#_0 e'_0$, and therefore $e_0 \#'_0 e'_0$.
If $f(e'_0) \#'_1 f(e_0)$ then obviously $e'_0 \#'_0 e_0$, implying $e_0 \#'_0 e_0$.
If there exists some $e_1 \in E_1$ such that $e_1 <_1 f(e'_0)$ and $e_1 \#'_1 f(e''_0)$, then there exists an $e''_0 \in E_0$ such that $f(e''_0) = e_1$ and $e''_0 <_0 e'_0$. This implies $e''_0 \#'_0 e_0$, and therefore $e_0 \#'_0 e'_0$.
3. Suppose $f(e_0) = f(e'_0)$ and $e'_0 \neq e_0$. Then, by condition 5 of an RPES morphism, $e_0 \#_0 e'_0$, and therefore $e_0 \#' e'_0$

Having shown that Φ_p maps from **RPES** to **PES**, the rest of the proof is similar to previous functors. □

While $\Phi_p(P_p(\mathcal{E})) = \mathcal{E}$ if $\mathcal{E}$ is a PES, $P_p(\Phi_p(\mathcal{E})) = \mathcal{E}$ only if $\mathcal{E}$ is an RPES with no reversible events [19]. However, Φ_p and P_p do form an adjunction. This means that for any RPES $\mathcal{E}$ there exists a natural transformation $\epsilon : P_p(\Phi_p(\mathcal{E})) \to \mathcal{E}$, and for any PES $\mathcal{E}'$, if there exists an RPES morphism $f : P_p(\mathcal{E}') \to \mathcal{E}$ then there exists a morphism $\overline{f} : \mathcal{E}' \to \Phi_p(\mathcal{E})$ such that $\epsilon \circ P_p(\overline{f}) = f$. In our case ϵ is the identity function on events and $\overline{f} = f$. Hence we get Proposition 4.16.

Proposition 4.16. $P_p \dashv \Phi_p$.

Example 4.17. For an example of how the functors C_{rp} and Φ_p work, we look at the RPES $\mathcal{E} = (E, F, <, \#, \prec, \triangleright)$ where $E = \{a, b, c\}$, $F = \{a, b\}$, $a < c$, $a \# b$, $a \prec \underline{a}$, $b \prec \underline{b}$, $c \prec \underline{a}$, and $c \triangleright \underline{b}$. This means that $\Phi_p(\mathcal{E}) = (E, <, \#')$ where $c \#' b$. The corresponding CSs $C_{rp}(\mathcal{E})$ and $C_p(\Phi_p(\mathcal{E}))$ can be seen in Fig. 12.

Notice that not only do the transitions involving reversal disappear, but because of hereditary conflict, so does the configuration $\{b, c\}$. Note also that $\{b, c\}$ was not forwards-reachable.

Since most reversible models restrict themselves to causal reversibility where actions cannot reverse if they have caused other actions, unlike RPESs which may allow other forms of reversibility as well, we define subcategories **crRPES** and **CRPES** of **RPES**, consisting of *cause-respecting* and *causal* RPESs as described in Definition 4.18 and the RPES morphisms between them.

Definition 4.18 *(CRPES and crRPES [19]).*

1. A *cause-respecting* RPES (crRPES) $\mathcal{E} = (E, F, <, \#, \prec, \triangleright)$ is an RPES such that for all $e \in E$ and $e' \in F$, if $e' < e$ then $e \triangleright \underline{e'}$. In other words, all causation is sustained causation as described in Definition 4.1.
2. A *causal* RPES (CRPES) $\mathcal{E} = (E, F, <, \#, \prec, \triangleright)$ is an RPES such that for all $e \in E$ and $e' \in F$, $e' < e$ if and only if $e \triangleright \underline{e'}$, and $e \prec \underline{e'}$ if and only if $e = e'$.

We can regard P_p as mapping to **CRPES** with empty set F. Consider the alternative mapping P'_p from [19], described as a functor below. This instead maps a PES to a CRPES where $F = E$, and the prevention is determined by the causality relation.

Definition 4.19 *(From **PES** to **CRPES**).* The functor $P'_p : \mathbf{PES} \to \mathbf{CRPES}$ is defined as

1. $P'_p((E, <, \#)) = (E, F, <, \#, \prec, \triangleright)$ where $F = E$, $e' \prec \underline{e'}$ for all $e' \in F$, and $e \triangleright \underline{e'}$ if and only if $e' < e$;
2. $P'_p(f) = f$.

Proposition 4.20. *P'_p is a functor.*

Proof. According to [19], for any PES $\mathcal{E}$, $P'_p(\mathcal{E})$ is a CRPES.

We show that given a PES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$, $f : P'_p(\mathcal{E}_0) \to P'_p(\mathcal{E}_1)$ is a CRPES morphism, satisfying the conditions of Definition 4.3: conditions 1, 2 and 5 are also conditions of a PES morphism and conditions 3, 4 and 6 are straightforward to prove.

Once established that P'_p is a mapping from **PES** to **CRPES**, the rest of the proof is similar to previous functors. □

We now show that P'_p forms a right adjoint of Φ_p between the categories **CRPES** and **PES**.

Proposition 4.21. $\Phi_p \dashv P'_p$.

Proof. For any CRPES $\mathcal{E} = (E, F, <, \#, \prec, \triangleright)$, we have $P'_p(\Phi_p(\mathcal{E})) = (E, E, <, \#', \prec', \triangleright')$. We define the unit $\eta : \mathcal{E} \to P'_p(\Phi_p(\mathcal{E}))$ as $\eta(e) = e$ for all $e \in E$.

We show that η is a CRPES morphism, satisfying the conditions of Definition 4.3:

1. Obviously $\{e \mid e < \eta(e')\} = \{\eta(e) \mid e < e'\}$.
2. Since all causation is sustained in $\mathcal{E}$, we have $\# = \#'$.
3. Suppose $e' \in F$ and $e \prec' \underline{\eta(e')}$. Then $e = e'$ and obviously $e \prec \underline{e'}$.
4. Suppose $e \in E$, $e' \in F$, and $e \triangleright' \underline{e'}$. Then $e' < e$, and therefore $e \triangleright \underline{e'}$.
5. No $e, e' \in E$ exist such that $\eta(e) = \eta(e')$ but $e \neq e'$.
6. Clearly $F \subseteq E$.

And since $\eta(e) = e$, clearly η is a natural transformation.

Let $\mathcal{E}$ be a CRPES, $\mathcal{E}'$ be a PES, and $g : \mathcal{E} \to P'_p(\mathcal{E}')$ be a CRPES morphism. If $\bar{g} : \Phi_p(\mathcal{E}) \to \mathcal{E}'$ uses the same mapping on events, then clearly $g(\mathcal{E}) = \Phi_p(\bar{g}) \circ \eta(\mathcal{E})$ and it is straightforward to show that $\bar{g}$ is a PES morphism. □

Remark 4.22. Since it is a right adjunction, P'_p preserves products. We therefore get a product in **CRPES** by mapping to PESs using Φ_p, finding their product [23], and mapping back using P'_p.

Note that this adjunction only exists between **PES** and **CRPES**, not **RPES**, and therefore this method for constructing products only works for CRPESs.

5. Reversible asymmetric event structures

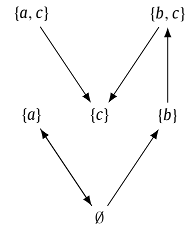
As we did with PESs, we will now add reversibility to AESs. *Reversible asymmetric event structures* (Definition 5.1) consist of events, some of which may be reversible, as well as causation $\prec$ and precedence $\lhd$, similar to an AES, except that $\prec$ is no longer a partial order, and instead just well-founded. In addition, both work on the reversible events, similarly to the RPES. One possible use of RAESs could be to model a scenario where an error causes an entire computation to reverse and start over, as illustrated by Example 5.2. We define the category of RAESs, along with coproducts (Definition 5.5), and extend the mappings from AESs and RPESs to RAESs, and from RAESs to AESs and CSs defined by [19] into functors. We show that the functors between RAESs and AESs form an adjunction (Proposition 5.14) similar to the one seen in Proposition 4.16.

Definition 5.1 *(RAES [19]).* A *reversible asymmetric event structure* (RAES) is a quadruple $\mathcal{E} = (E, F, \prec, \lhd)$ where E is the set of events and

1. $F \subseteq E$ is the set of reversible events;
2. $\lhd \subseteq (E \cup \underline{F}) \times E$ is the irreflexive precedence relation;
3. $\prec \subseteq E \times (E \cup \underline{F})$ is the causation relation, which is irreflexive and well-founded, such that for all $\alpha \in E \cup \underline{F}$, $\{e \in E \mid e \prec \alpha\}$ is finite and has no $\lhd$-cycles;
4. for all $e \in F$, $e \prec \underline{e}$;

$\mathcal{E} = (E, F, \prec, \lhd)$ where:
$E = \{a, b, c\}$
$F = \{a, b\}$
$b \prec c$, $c \prec \underline{b}$
$a \rhd b$, $b \rhd a$, $c \rhd a$, $c \rhd b$

(a) RAES $\mathcal{E}$



(b) CS derived from $\mathcal{E}$

Fig. 13. Example of an RAES and the corresponding CS.

5. for all $e \in E$ and $\alpha \in E \cup \underline{F}$ if $e \prec \alpha$ then not $e \rhd \alpha$;
6. $e \prec\!\!\prec e'$ implies $e \lhd e'$, where $e \prec\!\!\prec e'$ means that $e \prec e'$ and if $e \in F$ then $e' \rhd \underline{e}$;
7. $\prec\!\!\prec$ is transitive;
8. if $e \mathbin{\#} e'$ and $e \prec\!\!\prec e''$ then $e'' \mathbin{\#} e'$, where $\# = \lhd \cap \rhd$.

Example 5.2. Recall Example 1.2 where we wished to describe a computation, $e_0, e_1, e_2, \ldots$, which can be stopped and forced to undo itself causally by an error event. To model this as an RAES we need for $i \geq 0$: $e_i \prec e_{i+1}$, $e_{i+1} \rhd \underline{e_i}$, $\mathsf{error} \prec \underline{e_i}$, and $e_i \rhd \underline{\mathsf{error}}$.

We create a category **RAES** by defining RAES morphisms in Definition 5.3. This definition is obviously nearly identical to that of an AES morphism, with the added condition that, like in the RPES morphism, reversible events can only synchronise with other reversible events.

Definition 5.3 *(RAES morphism).* Let $\mathcal{E}_0 = (E_0, F_0, \prec_0, \lhd_0)$ and $\mathcal{E}_1 = (E_1, F_1, \prec_1, \lhd_1)$ be RAESs. An RAES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$ is a partial function $f : E_0 \to E_1$ such that

1. for all $e^* \in E_0 \cup \underline{F_0}$, if $f(e) \neq \bot$ then $\{e_1 \mid e_1 \prec_1 f(e^*)\} \subseteq \{f(e') \mid e' \prec_0 e^*\}$;
2. for all $e \in E_0$ and $e'^* \in E_0 \cup \underline{F_0}$, if $f(e) \neq \bot \neq f(e'^*)$ and $f(e'^*) \lhd_1 f(e)$ then $e'^* \lhd_0 e$;
3. for all $e, e' \in E_0$, if $f(e) = f(e') \neq \bot$ and $e \neq e'$ then $e \mathbin{\#_0} e'$;
4. $f(F_0) \subseteq F_1$.

Proposition 5.4. ***RAES**, where RAESs are objects and RAES morphisms are arrows, is a category.*

Proof. Similar to Proposition 4.4. □

We construct a coproduct of RAESs in Definition 5.5, in much the same way as we did for RPESs.

Definition 5.5 *(RAES coproduct).* Given RAESs $\mathcal{E}_0 = (E_0, F_0, \prec_0, \lhd_0)$ and $\mathcal{E}_1 = (E_1, F_1, \prec_1, \lhd_1)$, we can construct $\mathcal{E}_0 + \mathcal{E}_1 = (E, F, \prec, \lhd)$ with injections i_0 and i_1 where:

1. $E = \{(0, e) \mid e \in E_0\} \cup \{(1, e) \mid e \in E_1\}$;
2. $F = \{(0, e) \mid e \in F_0\} \cup \{(1, e) \mid e \in F_1\}$;
3. $(j, e) \prec (j', e')^*$ iff $j = j'$ and $e \prec_j e'^*$;
4. $(j, e) \rhd (j', e')^*$ iff $j \neq j'$ or $e \rhd_j e'^*$;
5. for $e \in E_j$, $i_j(e) = (j, e)$ for $j \in \{0, 1\}$.

Proposition 5.6. *Given RAESs $\mathcal{E}_0$ and $\mathcal{E}_1$, $\mathcal{E}_0 + \mathcal{E}_1$ is their coproduct in **RAES**.*

Proof. Let $\mathcal{E}_0 = (E_0, F_0, \prec_0, \lhd_0)$, $\mathcal{E}_1 = (E_1, F_1, \prec_1, \lhd_1)$, and $\mathcal{E}_0 + \mathcal{E}_1 = \mathcal{E} = (E, F, \prec, \lhd)$. Then clearly $\mathcal{E}$ is an RAES, and i_0 and i_1 as in Definition 5.5 are morphisms, and so we simply need to prove that if there exists an RAES $\mathcal{E}_2 = (E_2, F_2, \prec_2, \rhd_2)$ and morphisms $f_0 : \mathcal{E}_0 \to \mathcal{E}_2$ and $f_1 : \mathcal{E}_1 \to \mathcal{E}_2$, then there exists a unique RAES morphism $f : \mathcal{E} \to \mathcal{E}_2$ such that $i_0 \circ f = f_0$ and $i_1 \circ f = f_1$.

Since $E_0 + E_1$, i_0, and i_1 comprise a coproduct in the category of sets and partial functions, f must be unique.

We define f as $f((j, e)) = f_j(e)$. It is clear that $i_0 \circ f = f_0$ and $i_1 \circ f = f_1$. So it remains to show that f is an RAES morphism:

1. If $e_2 \prec_2 f(e)^*$ then there exists a $j \in \{0, 1\}$ and an $e_j^* \in E_j \cup \underline{F_j}$ such that $e^* = (j, e_j)^*$ and $e_2 \prec_2 f_j(e_j)^*$. Since f_j is a morphism, there must exist an $e'_j \in E_j$ such that $f_j(e'_j) = e_2$ and $e'_j \prec_j e_j^*$. Hence $(j, e'_j) \prec e^*$.

2. If $f(e')^* \lhd_2 f(e)$ then there must exist a $j \in \{0,1\}$ and $e_j \in E_j$ and $e_j'^* \in E_j \cup \underline{F_j}$ such that $f_j(e_j) = f(e)$, $f_j(e_j') = f(e')$ and $e_j'^* \lhd_j e_j$. Since $e_j'^* \lhd_j e_j$, $(j, e_j')^* \lhd (j, e_j)$.
3. If $f(e) = f(e') \neq \bot$ then $e = (j, e_j)$ and $e' = (j', e_{j'})$ and $f_j(e_j) = f(e) = f(e') = f_{j'}(e_{j'})$.
   If $j = j'$ then $e \neq e'$ implies $e_j \neq e_{j'}$. Hence $e_j \#_j e_{j'}$ and therefore $e \# e'$.
   If $j \neq j'$ then obviously $e \# e'$.
4. For $j \in \{0,1\}$, if $(j, e_j) \in F$ then $e_j \in F_j$, and $f_j(F_j) \subseteq F_2$. □

In order to model RAESs as CSs, [19] defined a mapping, C_{ra}. C_{ra} resembles C_{rp}, though as with AES configurations, the requirement of conflict-freedom is replaced with $\lhd$ being well-founded. Like C_{rp}, C_{ra} preserves morphisms, and is a functor. An example of applying C_{ra} to an RAES can be seen in Fig. 13.

Definition 5.7 *(From **RAES** to **CS**).* The functor $C_{ra} : \mathbf{RAES} \to \mathbf{CS}$ is defined as:

1. $C_{ra}((E, F, \prec, \lhd)) = (E, F, \mathsf{C}, \to)$ where $\mathsf{C} = \{X \subseteq E \mid \lhd \text{ is well-founded on } X\}$ and $X \xrightarrow{A \cup \underline{B}} Y$ if
   (a) $X, Y \in \mathsf{C}$, $A \subseteq E$, and $B \subseteq F$;
   (b) $Y = (X \setminus B) \cup A$;
   (c) $A \cap X = \emptyset$;
   (d) $B \subseteq X$;
   (e) for every $e \in A$, if $e' \prec e$ then $e' \in X \setminus B$;
   (f) for every $e \in A$, if $e' \rhd e$ then $e' \notin X \cup A$;
   (g) for every $e \in B$, if $e' \prec \underline{e}$ then $e' \in X \setminus (B \setminus \{e\})$;
   (h) for every $e \in B$, if $e' \rhd \underline{e}$ then $e' \notin X \cup A$.
2. $C_{ra}(f) = f$.

Proposition 5.8. *C_{ra} is a functor from **RAES** to **CS**.*

Proof. [19] showed that for an RAES $\mathcal{E}$, $C_{ra}(\mathcal{E})$ is a CS.

Let $\mathcal{E}_0 = (E_0, F_0, \prec_0, \lhd_0)$ and $\mathcal{E}_1 = (E_1, F_1, \prec_1, \lhd_1)$ be RAESs, and let $C_{ra}(\mathcal{E}_0) = (E_0, F_0, \mathsf{C}_0, \to_0)$ and $C_{ra}(\mathcal{E}_1) = (E_1, F_1, \mathsf{C}_1, \to_1)$. Suppose $f : \mathcal{E}_0 \to \mathcal{E}_1$ is an RAES morphism. Then $f : C_{ra}(\mathcal{E}_0) \to C_{ra}(\mathcal{E}_1)$ is a CS morphism because it satisfies the conditions of Definition 3.3:

1. If $X \in \mathsf{C}_0$, then $\lhd_0$ is well-founded on X. Since, by definition of an RAES morphism, $f(e_0) \lhd_1 f(e_0') \Rightarrow e_0 \lhd_0 e_0'$, this implies that $\lhd_1$ is well-founded on $f(X)$, and therefore $f(X) \in \mathsf{C}_1$.
2. $f(X) \xrightarrow{f(A) \cup f(\underline{B})}_1 f(Y)$ according to Definition 5.7:
   (a) Since $X, Y \in \mathsf{C}_0$, this is implied by the first item.
   (b) Obvious, since $Y = (X \setminus B) \cup A$ and $B \subseteq X$.
   (c) $A \cap X = \emptyset$, and by definition of an RAES morphism, for all $e_0, e_0' \in E_0$, if $f(e_0) = f(e_0') \neq \bot$ and $e_0 \neq e_0'$ then $e_0 \#_0 e_0'$, implying $e_0, e_0' \notin (X \cup A)$, since $\lhd_0$ is well-founded on X, and for every $a_0 \in A$, if $c_0 \rhd_0 a_0$ then $c_0 \notin X \cup A$.
   (d) $B \subseteq X$.
   (e) For every $a_0 \in A$, if $c_0 \prec_0 a_0$ then $c_0 \in X \setminus B$, implying $f(c) \in f(X) \setminus f(B)$, and by definition of an RAES $\{e_1 \mid e_1 \prec_1 f(a_0)\} \subseteq \{f(e') \mid e' \prec_0 a_0\}$.
   (f) Since $a_1 \in f(A)$ there must exist an $a_0 \in A$ such that $f(a_0) = a_1$. If $c_1 \in f(X) \cup f(A)$ then there must exist a $c_0 \in X \cup A$ such that $f(c_0) = c_1$. Since $f(c_0) \rhd_1 f(a_0)$, by definition of an RAES morphism $c_0 \rhd_0 a_0$, implying $c_0 \notin X \cup A$ which is a contradiction.
   (g) Similar to above.
   (h) Similar to above.
3. By definition of an RAES morphism, $e_0 \#_0 e_0'$, and since $\lhd_0$ is well-founded on all $X \in \mathsf{C}_0$, $e_0, e_0' \notin X$.

Having established that C_{ra} is a mapping, the rest of the proof is similar to previous functors. □

In order to create functors from **RAES** to **AES** (Definition 5.10), based on a mapping from [19], we need to be able to split the causality and precedence relations into forwards and reverse parts.

Definition 5.9. Given an RAES $\mathcal{E} = (E, F, \prec, \lhd)$, let

1. $\prec_E = \prec \cap (E \times E)$ and $\prec_F = \prec \cap (E \times \underline{F})$;
2. $\lhd_E = \lhd \cap (E \times E)$ and $\lhd_F = \lhd \cap (E \times \underline{F})$.

We also define functors from **RPES** and **AES** to **RAES**, also based on mappings from [19]. To turn an RAES into an AES, we need to remove events with $< \cup \lhd$-cycles in their causes, as these are not allowed in the definition of an AES. Conditions 6-8 of Definition 5.1 of an RAES also only apply to sustained causation, while in an AES, all causation is sustained. We therefore expand the asymmetric conflict and causation relations to fulfil those conditions for all causation.

Definition 5.10 *(From **RAES** to **AES**).* The functor $\Phi_a : \mathbf{RAES} \to \mathbf{AES}$ is defined as:

1. $\Phi_a((E, F, \prec, \lhd)) = (E', <', \lhd')$ where E' is E where any event e that has a $\lhd_E \cup \prec_E$-cycle in $\{e' \in E \mid e' = e \text{ or } e' \prec_E^+ e\}$ has been removed, $<' = \prec_E^+ \cap (E' \times E')$, and $\lhd'$ is $\lhd_E \cup <_E$ closed under conflict heredity, using the rules
$$\frac{a \lhd b}{a \lhd' b} \quad \frac{a <' b}{a \lhd b} \quad \frac{a \,\#'\, b <' c}{a \,\#'\, c}$$
2. $\Phi_a(f) = f \upharpoonright E'$.

Proposition 5.11. *Φ_a is a functor from **RAES** to **AES**.*

Proof. [19] showed that for an RAES $\mathcal{E}$, $\Phi_a(\mathcal{E})$ is an AES.

If $f : \mathcal{E}_0 \to \mathcal{E}_1$ is an RAES morphism then $\Phi_a(f) = f' : \Phi_a(\mathcal{E}_0) \to \Phi_a(\mathcal{E}_1)$ is a AES morphism because it satisfies the conditions of Definition 2.12:

1. Suppose $e_1 <'_1 f'(e_0)$. Then there must exist some sequence $e_1 \prec_1 e_{1_1} \prec \cdots \prec_1 e_{1_k} \prec_1 f(e_0)$ for some $0 \le k$. Since f is an RAES morphism, there must exists a corresponding sequence $e'_0 \prec_0 e_{0_1} \prec_0 \cdots \prec_0 e_{1_k} \prec_0 e_0$ such that $f(e'_0) = e_1$ and $f(e_{0_i}) = e_{1_i}$. This implies that $e'_0 <'_0 e_0$.
2. Suppose $f'(e_0) \lhd'_1 f'(e'_0)$. Then either $f'(e_0) \lhd_1 f'(e'_0)$, $f'(e_0) <'_1 f'(e'_0)$, or there exists some $e_1 \in E_1$ such that $e_1 <'_1 f'(e'_0)$ and $f'(e_0) \,\#'_1\, e_1$.
If $f'(e_0) \rhd_1 f'(e'_0)$ then obviously $e_0 \rhd_0 e'_0$, and therefore $e_0 \rhd'_0 e'_0$.
If $f'(e'_0) <'_1 f'(e_0)$ then there exists e''_0 such that $f'(e''_0) = f'(e'_0)$ and $e''_0 <'_0 e_0$. If $e''_0 = e'_0$ then obviously $e'_0 <'_0 e_0$, implying $e_0 \rhd'_0 e'_0$.
If there exists some $e_1 \in E_1$ such that $e_1 <'_1 f'(e'_0)$ and $f'(e_0) \,\#'_1\, e_1$, then there exists an $e''_0 \in E_0$ such that $f'(e''_0) = e_1$ and $e''_0 <'_0 e'_0$. Hence $e_0 \,\#'_0\, e''_0$, and therefore $e_0 \lhd'_0 e'_0$.
3. Suppose $f(e_0) = f(e'_0)$ and $e_0 \neq e'_0$. Then according to the definition of an RAES morphism, $e_0 \rhd_0 e'_0$, and therefore $f(e_0) \rhd'_0 f(e'_0)$.

Once established that Φ_a is a mapping from **RAES** to **AES**, the rest of the proof is similar to previous functors. □

Definition 5.12 *(From **AES** to **RAES**).* The functor $P_a : \mathbf{AES} \to \mathbf{RAES}$ is defined as:

1. $P_a((E, <, \lhd)) = (E, \emptyset, <, \lhd)$;
2. $P_a(f) = f$.

Proposition 5.13. *P_a is a functor from **AES** to **RAES**.*

Proof. [19] showed that for an AES $\mathcal{E}$, $P_a(\mathcal{E})$ is an AES.

If $f : \mathcal{E}_0 \to \mathcal{E}_1$ is an AES morphism then $f : P_a(\mathcal{E}_0) \to \Phi_a(\mathcal{E}_1)$ is an RAES morphism because it satisfies the conditions of Definition 5.3 because $F_0 = F_1 = \emptyset$ and conditions 1, 2, and 3 are implied by Definition 2.12.

Having established that P_a is a mapping from **AES** to **RAES**, the rest of the proof is similar to previous functors. □

The relationship between Φ_a and P_a mimics that of Φ_p and P_p in two ways: (1) $\Phi_a(P_a(\mathcal{E})) = \mathcal{E}$ if $\mathcal{E}$ is a AES, but $P_a(\Phi_a(\mathcal{E})) = \mathcal{E}$ only if $\mathcal{E}$ is an RAES with no reversible events [19], and (2) they form an adjunction, as described in Proposition 5.14.

Proposition 5.14. *$P_a \dashv \Phi_a$.*

Proof. The proof is similar to Proposition 4.16, also using counit $\epsilon(e) = e$ and $\bar{f} = f$. □

To map an RPES to an RAES, we must combine the forwards and reverse causation into one relation. We also need to define an asymmetric conflict relation. In the functor from PES to AES, this was done by adding the causation to the conflict, but here we only add the sustained causation, in order to fulfil Condition 6 of Definition 5.1. We need to add the prevention as well.

Definition 5.15 *(From **RPES** to **RAES**).* The functor $A_r : \mathbf{RPES} \to \mathbf{RAES}$ is defined as:

1. $A_r((E, F, <, \#, \prec, \rhd)) = (E, F, < \cup \prec, \ll \cup \# \cup \lhd)$;
2. $A_r(f) = f$.

Proposition 5.16. *A_r is a functor from **RPES** to **RAES**.*

Proof. [19] showed that for an RPES $\mathcal{E}$, $A_r(\mathcal{E})$ is an RAES.

If $f : \mathcal{E}_0 \to \mathcal{E}_1$ is an RPES morphism then $A_r(f) = f : A_r(\mathcal{E}_0) \to A_r(\mathcal{E}_1)$ is an RAES morphism because it satisfies the conditions of Definition 5.3:

1. According to the definition of A_r, $\prec'_0 = <_0 \cup \prec_0$ and $\prec'_1 = <_1 \cup \prec_1$, and according to the definition of an RPES morphism for all $e \in E_0$, if $f(e) \neq \bot$ then $\{e_1 \mid e_1 <_1 f(e)\} \subseteq \{f(e') \mid e' <_0 e\}$, and for all $e \in F_0$, if $f(e) \neq \bot$ then $\{e_1 \mid e_1 \prec_1 \underline{f(e)}\} \subseteq \{f(e') \mid e' \prec_0 \underline{e}\}$.
2. According to the definition of A_r, $\lhd'_0 = \ll_0 \cup \#_0 \cup \lhd_0$ and $\lhd'_1 = \ll_1 \cup \#_1 \cup \lhd_1$, and according to the definition of an RPES morphism for all $e, e' \in E_0$, if $f(e) \neq \bot \neq f(e')$ and $f(e) \#_1 f(e')$ then $e \#_0 e'$, and for all $e \in E_0$ and $e' \in F_0$, if $f(e) \neq \bot \neq f(e')$ and $f(e) \rhd_1 \underline{f(e')}$ then $e \rhd_0 \underline{e'}$.
   If $f(e') \ll_1 f(e)$ then $f(e') <_1 f(e)$ and either $f(e') \notin F_1$ or $f(e) \rhd \underline{f(e')}$. Since $\{e_1 \mid e_1 <_1 f(e)\} \subseteq \{f(e') \mid e' <_0 e\}$, $f(e') <_1 f(e)$ implies that there exists $e'' \in E_0$ such that $f(e'') = f(e')$ and $e'' <_0 e$. If $f(e') \notin F_1$ then, since $f(F_0) \subseteq F_1$, $e' \notin F_0$, and $e'' \ll_0 e$. If $f(e) \rhd_1 f(e')$ then, as shown above, $e \rhd_0 e'$ and again $e'' \ll_0 e$. If $e'' = e'$ then this is enough. Otherwise, since $\#_0$ is hereditary with respect to $\ll_0$ and $e'' \#_0 e'$, $e'' \#_0 e$.
3. This is item 5 of the definition of an RPES morphism.
4. This is item 6 of the definition of an RPES morphism.

Once established that A_r is a mapping from **RPES** to **RAES**, the rest of the proof is similar to previous functors. □

We define subcategories **crRAES** and **CRAES** consisting of the cause-respecting and causal RAESs defined in Definition 5.17 and the RAES morphisms between them.

Definition 5.17 *(CRAES and crRAES [19]).*

1. A *cause-respecting* RAES (crRAES) $\mathcal{E} = (E, F, \prec, \lhd)$ is an RAES such that for all $e \in E$ and $e' \in F$, if $e' \prec e$ or $e \rhd e'$ then $e \rhd \underline{e'}$.
2. A *causal* RAES (CRAES) $\mathcal{E} = (E, F, \prec, \lhd)$ is an RAES such that for all $e \in E$ and $e' \in F$, $e \rhd \underline{e'}$ if and only if $e' \prec e$ or $e \rhd e'$, and $e \prec \underline{e'}$ if and only if $e = e'$.

The functor P_a maps AESs to CRAESs with no reversible events, and we define an alternative functor, P'_a, based on a mapping from [19], which maps AESs to CRAESs where all events are reversible.

Definition 5.18 *(From **AES** to **CRAES**).* The functor P'_a is defined as

1. $P'_a((E, <, \lhd)) = (E, F, \prec, \lhd')$ where $F = E$, $e \prec e'^*$ if $e < e'^*$ or $e'^* = \underline{e}$, and $e^* \lhd' e'$ if $e \lhd e'$;
2. $P'_a(f) = f$.

Proposition 5.19. *P'_a is a functor from **AES** to **CRAES**.*

Proof. According to [19], given an AES $\mathcal{E}$, $P'_a(\mathcal{E})$ is a CRAES.

Given an AES morphism, $f : \mathcal{E}_0 \to \mathcal{E}_1$, it is straightforward to show that $f : P'_a(\mathcal{E}_0) \to P'_a(\mathcal{E}_1)$ is a CRAES morphism, satisfying the conditions of Definition 5.3.

Having established that P'_a is a mapping from **AES** to **CRAES**, the rest of the proof is similar to previous functors. □

We show that P'_a forms a right adjoint of Φ_a between the categories **CRAES** and **AES**.

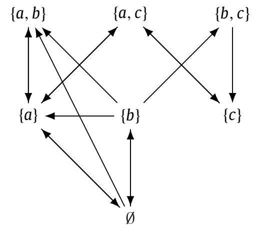
Proposition 5.20. $\Phi_a \dashv P'_a$.

Proof. For any CRAES $\mathcal{E} = (E, F, \prec, \lhd)$, we have $P'_a(\Phi_a(\mathcal{E})) = (E', E', \prec', \rhd')$. We define the unit $\eta : \mathcal{E} \to P'_p(\Phi_p(\mathcal{E}))$ as $\eta(e) = e$ for all $e \in E$. We show that η is a CRAES morphism, satisfying the conditions of Definition 5.3:

1. Obviously $\{e \mid e < \eta(e')\} = \{\eta(e) \mid e < e'\} \cap E'$ and if $e' \in F$ and $e \prec' \underline{\eta(e')}$, then $e = e'$ and obviously $e' \prec \underline{e}$.

$\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ where
$E = F = \{a, b, c\}$
$\mathsf{Con} = \{\emptyset, \{a\}, \{b\}, \{a, b\}, \{a, c\}, \{b, c\}\}$
$\emptyset \oslash \emptyset \vdash a$, $\emptyset \oslash \{c\} \vdash b$,
$\{a\} \oslash \{b\} \vdash c$, $\{b\} \oslash \{a\} \vdash c$
$\{b\} \oslash \emptyset \vdash \underline{b}$, $\{a\} \oslash \{b\} \vdash \underline{a}$,
$\{a, c\} \oslash \emptyset \vdash \underline{c}$

(a) RES $\mathcal{E}$



(b) The CS derived from $\mathcal{E}$

Fig. 14. Example of an RES and the corresponding CS.

2. Suppose $e \in E$, $e'^* \in E \cup \underline{F}$, and $e \rhd' e'^*$. If $e'^* = e'$ then clearly $e \rhd e'$. And if $e'^* = \underline{e'}$ then $e' \lhd e$, and since $\mathcal{E}$ is a CRAES, $e \rhd \underline{e'}$.
3. No $e, e' \in E$ exist such that $\eta(e) = \eta(e')$ but $e \neq e'$.
4. Clearly $F \subseteq E$.

Furthermore, since $\eta(e) = e$, η is clearly a natural transformation.

Let $\mathcal{E}_0$ be a CRAES, $\mathcal{E}_1$ be an AES, and $g : \mathcal{E}_0 \to P'_a(\mathcal{E}_1)$ be a CRAES morphism. If $\bar{g} : \Phi_a(\mathcal{E}_0) \to \mathcal{E}_1$ is g restricted to E'_0, then clearly $g(\mathcal{E}_0) = \Phi_a(\bar{g}) \circ \eta(\mathcal{E}_0)$ and it is straightforward to show that $\bar{g}$ is an AES morphism.

1. Suppose $e_1 <_1 f(e)$. Then there exists an event $e_0 \in E_0$ such that $e_0 \prec_0 e$ and $f(e_0) = e_1$. We therefore need to show that $e_0 \in E'_0$. As $\mathcal{E}_0$ is a CRAES, $\lll_0 = \prec_{0_E}$, and therefore $\prec_{0_E} = \prec^+_{0_E}$. Hence any $\prec \cup \lhd$-cycle in $\{e' \in E \mid e' = e_0 \text{ or } e' \prec^+_E e\}$ would be a $\lhd$-cycle in $\{e' \mid e' = e_0 \text{ or } e' \prec e_0\}$, which is impossible.
2. Suppose $f(e) \lhd_1 f(e')$. Then $f(e) \lhd_1 f(e')$, and therefore $e \lhd_{0_E} e'$.
Suppose $f(e) = f(e') \neq \bot$. Then $e \# e'$ by definition of an RAES morphism. $\square$

6. Reversible event structures

The last kind of event structure to which we add reversibility is the general event structure. The *reversible (general) event structure* (Definition 6.1) differs from the general event structure, not only by allowing the reversal of events, but also by including a preventing set in the enabling relation, so that $X \oslash Y \vdash e$ means e is enabled in configurations that include all the events of X but none of the events of Y. Example 6.2 shows how RESs can be used to model aspects of the reversible π-calculus that RPESs and RAESs cannot. We show that, unlike in the forward-only setting, most RAESs can be modelled by RESs by using prevention (Theorem 6.23), though the fact that we can only describe consistency of finite sets of events is still an issue. We also formulate a subcategory of CSs, called finitely enabled CSs (Definition 6.11), and create functors between these and RESs, called C_r (Definition 6.10) and R (Definition 6.14), which form an adjunction (Theorem 6.17) and are inverses on a subset of finitely enabled CSs (Theorem 6.16).

Definition 6.1 *(RES [21]).* A *reversible event structure* (RES) is a triple $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ where E is the set of events, $F \subseteq E$ is the set of reversible events, $\mathsf{Con} \subseteq_{\text{fin}} 2^E$ is the consistency relation, which is downwards-closed, $\vdash \subseteq \mathsf{Con} \times 2^E \times (E \cup \underline{F})$ is the enabling relation, and

1. if $X \oslash Y \vdash e^*$ then $(X \cup \{e\}) \cap Y = \emptyset$;
2. if $X \oslash Y \vdash \underline{e}$ then $e \in X$;
3. (weakening) if $X \oslash Y \vdash e^*$, $X \subseteq X' \in \mathsf{Con}$, and $X' \cap Y = \emptyset$ then $X' \oslash Y \vdash e^*$.

Note that in examples we will leave out enablings which are implied by the weakening condition. See Fig. 14 for an example of an RES and the corresponding CS.

Example 6.2. Recall Example 1.3 where we modelled the reversible π-calculus process $(\nu n)(\bar{a}\langle n\rangle \mid \bar{b}\langle n\rangle \mid c(x))$. This can be achieved with an RES with a set of events $\{\bar{a}\langle n\rangle, \bar{b}\langle n\rangle\} \cup \{c(x) \mid x \text{ is a name}\}$ with the consistent sets being any set not containing more than one event from $\{c(x) \mid x \text{ is a name}\}$. The enablings are as follows:

$\emptyset \oslash \emptyset \vdash \bar{a}\langle n\rangle$	$\emptyset \oslash \emptyset \vdash \bar{b}\langle n\rangle$
$\emptyset \oslash \emptyset \vdash c(x)$ for any name $x \neq n$	
$\{\bar{a}\langle n\rangle\} \oslash \{c(x) \mid x \neq n\} \vdash c(n)$	$\{\bar{b}\langle n\rangle\} \oslash \{c(x) \mid x \neq n\} \vdash c(n)$
$\{\bar{a}\langle n\rangle, \bar{b}\langle n\rangle, c(n)\} \oslash \emptyset \vdash \underline{\bar{b}\langle n\rangle}$	$\{\bar{b}\langle n\rangle, \bar{a}\langle n\rangle, c(n)\} \oslash \emptyset \vdash \underline{\bar{a}\langle n\rangle}$
$\{\bar{b}\langle n\rangle\} \oslash \{c(n)\} \vdash \underline{\bar{b}\langle n\rangle}$	$\{\bar{a}\langle n\rangle\} \oslash \{c(n)\} \vdash \underline{\bar{a}\langle n\rangle}$
$\{c(x)\} \oslash \emptyset \vdash \underline{c(x)}$ for any name x	

This RES cannot be represented by an RPES or an RAES, as they cannot express that $\overline{b}\langle n\rangle$ can reverse if $\overline{a}\langle n\rangle$ and $c(n)$ are in the configuration, but not if $c(n)$ is present without $\overline{a}\langle n\rangle$.

We next define a notion of morphism for RESs in Definition 6.3. With the exception of the requirements regarding preventing sets, it is identical to the definition of an ES morphism. We treat the preventing set similarly to (asymmetric) conflict in PES, AES, RPES, and RAES morphisms, requiring the events of Y_1 to be either synchronised with an event from Y_0 or no event at all.

The reason we do not say that if $f(e) \neq \perp$ and $X \oslash Y \vdash_0 e^*$ then $f(X) \oslash f(Y) \vdash_1 f(e)^*$ is that we want our definition of an RES morphism to correspond with that of an RAES morphism, and we will use $\{e' \mid e' \rhd e\}$ to define our preventing sets when translating. Since RAES morphisms allow $e_1 \rhd f(e)$ if no e_0 exists such that $f(e_0) = e_1$, we have to express the same in our definition of RES morphisms.

Definition 6.3 *(RES morphism).* Let $\mathcal{E}_0 = (E_0, F_0, \mathsf{Con}_0, \vdash_0)$ and $\mathcal{E}_1 = (E_1, F_1, \mathsf{Con}_1, \vdash_1)$ be RESs. An RES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$ is a partial function $f : E_0 \to E_1$ such that:

1. for all $e^* \in E_0 \cup \underline{F_0}$, if $f(e) \neq \perp$ and $X_0 \oslash Y_0 \vdash_0 e^*$ then there exists a $Y_1 \subseteq E_1$ such that for all $e_0 \in E_0$, if $f(e_0) \in Y_1$ then $e_0 \in Y_0$ and $\overline{f}(X_0) \oslash Y_1 \vdash_1 f(e)^*$.
2. for any $X_0 \in \mathsf{Con}_0$, $f(X_0) \in \mathsf{Con}_1$.
3. for all $e, e' \in E_0$, if $f(e) = f(e') \neq \perp$ and $e \neq e'$ then no $X \in \mathsf{Con}_0$ exists such that $e, e' \in X$.

Proposition 6.4. ***RES**, where RESs are objects and RES morphisms are arrows, is a category.*

Proof. Since RES morphisms are partial functions they are obviously associative, and $f(e) = e$ is the identity arrow. Then we only need to prove that given RESs $\mathcal{E}_0$, $\mathcal{E}_1$, and $\mathcal{E}_2$ and morphisms $f : \mathcal{E}_0 \to \mathcal{E}_1$ and $g : \mathcal{E}_1 \to \mathcal{E}_2$, $g \circ f : \mathcal{E}_0 \to \mathcal{E}_2$ is a morphism, satisfying the conditions in Definition 6.3:

1. Suppose $g(f(e)) \neq \perp$ and $X \oslash Y \vdash_0 e$. Then there exists an $e_1 \in E_1$ and an $e_2 \in E_2$ such that $f(e) = e_1$ and $g(e_1) = e_2$. It is clear from Definition 6.3 that there exists a $Y_1 \subseteq E_1$ such that for all $e'_0 \in E_0$ such that $f(e'_0) \in Y_1$, $e'_0 \in Y$, and $f(X) \oslash Y_1 \vdash_1 f(e)^*$, and therefore there exists a $Y_2 \subseteq E_2$ such that for all $e'_1 \in E_1$ such that $g(e'_1) \in Y_2$, $e'_1 \in Y_1$, and $g(f(X)) \oslash Y_2 \vdash_2 g(f(e))^*$.
2. Suppose $X_0 \in \mathsf{Con}_0$. Since f is a morphism, obviously $f(X_0) \in \mathsf{Con}_1$, and using the same argument, $g(f(X_0)) \in \mathsf{Con}_2$.
3. Suppose $g(f(e)) = g(f(e')) \neq \perp$ and $e \neq e'$. Then $f(e) = e_1$ and $f(e') = e'_1$ for some $e_1, e'_1 \in E_1$ such that $g(e_1) = g(e'_1) \neq \perp$. Therefore either $e_1 = e'_1$ or no $X_1 \in \mathsf{Con}_1$ exists such that $e_1, e'_1 \in X_1$.
   In the first case, since f is a morphism, by definition if $e \neq e'$ then no $X \in \mathsf{Con}_0$ exists such that $e, e' \in X$
   In the second, by definition for any $X_0 \in \mathsf{Con}_0$, $f(X_0) \in \mathsf{Con}_1$, and therefore there cannot exist any $X \in \mathsf{Con}_0$ such that $e, e' \in X$. □

We construct a coproduct of two RESs in Definition 6.5. Although this definition is a little different from those of RPES and RAES coproducts, the principle is very much the same. Similarly to having every event from E_j prevent every event from E_{1-j} in the RAES coproduct, we here put every event from E_j in the preventing set of every enabling of an event from E_{1-j}.

Definition 6.5 *(RES coproduct).* Given RESs $\mathcal{E}_0 = (E_0, \mathsf{Con}_0, \vdash_0)$ and $\mathcal{E}_1 = (E_1, \mathsf{Con}_1, \vdash_1)$, we can construct $\mathcal{E}_0 + \mathcal{E}_1 = (E, \mathsf{Con}, \vdash)$ with injections i_0 and i_1 where:

1. $E = \{(0, e) \mid e \in E_0\} \cup \{(1, e) \mid e \in E_1\}$);
2. for $e \in E_j$ and $j \in \{0, 1\}$ we have $i_j(e) = (j, e)$;
3. $X \in \mathsf{Con}$ iff $\exists X_0 \in \mathsf{Con}_0 . i_0(X_0) = X$ or $\exists X_1 \in \mathsf{Con}_1 . i_1(X_1) = X$;
4. $X \oslash Y \vdash (j, e)^*$ iff $\exists X_j, Y_j \in E_j$ such that $X_j \oslash Y_j \vdash e^*$, $i_j(X_j) = X$, $Y = i_j(Y_j) \cup (E \setminus i_j(E_j))$.

Proposition 6.6. *Given RESs $\mathcal{E}_0$ and $\mathcal{E}_1$, $\mathcal{E}_0 + \mathcal{E}_1$ is their coproduct in **RES**.*

Proof. Let $\mathcal{E}_0 = (E_0, \mathsf{Con}_0, \vdash_0)$, $\mathcal{E}_1 = (E_1, \mathsf{Con}_1, \vdash_1)$, and $\mathcal{E}_0 + \mathcal{E}_1 = \mathcal{E} = (E, \mathsf{Con}, \vdash)$. Then obviously $\mathcal{E}$ is an RES, and i_0 and i_1 are morphisms, so we simply need to prove that if there exists an RES $\mathcal{E}_2 = (E_2, F_2, \prec_2, \rhd_2)$ and morphisms $f_0 : \mathcal{E}_0 \to \mathcal{E}_2$ and $f_1 : \mathcal{E}_1 \to \mathcal{E}_2$, then there exists a unique RES morphism $f : \mathcal{E} \to \mathcal{E}_2$ such that $f \circ i_0 = f_0$ and $f \circ i_1 = f_1$.

Since $E_0 + E_1$, i_0, and i_1 comprise a coproduct in the category of sets and partial functions, f must be unique.

We define f as $f((j, e)) = f_j(e)$, giving us $f \circ i_0 = f_0$ and $f \circ i_1 = f_1$, and we prove that f is a morphism, satisfying the conditions of Definition 6.3:

1. Suppose $X \oslash Y \vdash e^*$ and $e = (j, e_j)$ for some $j \in \{0, 1\}$. Then there exist $X_j, Y_j \subseteq E_j$ such that $i_j(X_j) = X$, $Y = i_j(Y_j) \cup (E \setminus i_j(E_j))$, and $X_j \oslash Y_j \vdash e_j$. If $f(e) \neq \perp$, meaning $f_j(e_j) \neq \perp$, then there exists a $Y_2 \subseteq E_2$ such that for all $e_j \in E_j$ if $f_j(e_j) \in Y_2$ then $e_j \in Y_j$, and $f_j(X_j) \oslash Y_2 \vdash_2 f(e)^*$. In addition, for all $(j', e') \in E$, such that $j \neq j'$, $(j', e') \in Y$, implying that if $f_{j'}((j', e')) \in Y_2$ then $(j', e') \in Y$.

2. Suppose $X \in \mathsf{Con}$. Then there must exist $j \in \{0,1\}$ and $X_j \in \mathsf{Con}_j$ such that $i_j(X_j) = X$. Since $X_j \in \mathsf{Con}_j$, $f(X) = f_j(X_j) \in \mathsf{Con}_2$.
3. Suppose $f(e) = f(e') \neq \perp$. Then $e = (j, e_j)$ and $e' = (j', e_{j'})$ and $f_j(e_j) = f(e) = f(e') = f_{j'}(e_{j'})$.
   If $j \neq j'$ then obviously no $X \in \mathsf{Con}$ exists such that $e, e' \in X$.
   If $j = j'$ then $e \neq e'$ implies that $e_j \neq e_{j'}$, and since f_j is an RES morphism we get that no $X_j \in \mathsf{Con}_j$ exists such that $e_j, e_{j'} \in X_j$. For there to exist an $X \in \mathsf{Con}$ such that $e, e' \in X$, we need an $X'_j \in \mathsf{Con}_j$ such that $i(X'_j) = X$, which obviously does not exist. □

We also construct a product of RESs in Definition 6.7 based on the ES product described in [25]. This notion of product translates better to the reversible setting than ones used for PESs. In addition to adding reverse enablings, which are treated much the same as forward enablings, our product must also deal with preventing sets, which are not present in the product of ESs. Just as in the product of CSs, we can consider the product of RES a synchronisation of two RESs. Each enabling of an event, $X \otimes Y \vdash (e_0, *)$, must correspond to an enabling of the original event $X_0 \otimes Y_0 \vdash_0 e_0$, such that $\pi_0(X) = X_0$, but Y includes every e' such that $\pi_0(e') \in Y_0$. This means that we generate an enabling for each X for which this is true, all using the same Y, so $(e_0, *)$ is enabled as long as each e'_0 in X_0 has been involved in some synchronisation, but none of the e''_0 in Y have. Enablings of $(*, e_1)$ are created in a similar way, and enablings of (e_0, e_1) require this to hold for both. When creating enablings for (e_0, e_1), the weakening condition in Definition 6.1 ensures that we also get enablings where an event from X_0 synchronises with an event from $E_1 \setminus (X_1 \cup Y_1)$ or vice versa.

Definition 6.7 *(partially synchronous RES product).* Given RESs $\mathcal{E}_0 = (E_0, \mathsf{Con}_0, \vdash_0)$ and $\mathcal{E}_1 = (E_1, \mathsf{Con}_1, \vdash_1)$, we can construct $\mathcal{E}_0 \times \mathcal{E}_1 = (E, \mathsf{Con}, \vdash)$ with projections π_0 and π_1 where:

1. $E = E_0 \times_* E_1 = \{(e, *) \mid e \in E_0\} \cup \{(*, e) \mid e \in E_1\} \cup \{(e, e') \mid e \in E_0 \text{ and } e' \in E_1\}$;
2. for $(e_0, e_1) \in E$ and $j \in \{0,1\}$, $\pi_j((e_0, e_1)) = e_j$;
3. $X \in \mathsf{Con}$ if
   (a) $\pi_0(X) \in \mathsf{Con}_0$;
   (b) $\pi_1(X) \in \mathsf{Con}_1$;
   (c) for all $e, e' \in X$, if $\pi_0(e) = \pi_0(e') \neq *$ or $\pi_1(e) = \pi_1(e') \neq *$ then $e = e'$;
4. $X \otimes Y \vdash e^*$ if $X \in \mathsf{Con}$ and
   (a) if $\pi_0(e) \neq *$ then there exists $Y_0 \subseteq E_0$ such that $\pi_0(X) \otimes Y_0 \vdash \pi_0(e)^*$ otherwise $Y_0 = \emptyset$;
   (b) if $\pi_1(e) \neq *$ then there exists $Y_1 \subseteq E_1$ such that $\pi_1(X) \otimes Y_1 \vdash \pi_1(e)^*$ otherwise $Y_1 = \emptyset$;
   (c) $Y = \{e \mid \pi_0(e) \in Y_0 \text{ or } \pi_1(e) \in Y_1\}$;
   (d) if $e^* = \underline{e}$ then $e \in X$.

Proposition 6.8. *Given RESs $\mathcal{E}_0$ and $\mathcal{E}_1$, $\mathcal{E}_0 \times \mathcal{E}_1$ is a product in the category* ***RES****.*

Proof. Let $\mathcal{E}_0 = (E_0, F_0, \mathsf{Con}_0, \vdash_0)$, $\mathcal{E}_1 = (E_1, F_1, \mathsf{Con}_1, \vdash_1)$, and $\mathcal{E}_0 \times \mathcal{E}_1 = \mathcal{E} = (E, F, \mathsf{Con}, \vdash)$. Then it should be obvious that $\mathcal{E}$ is an RES and π_0 and π_1, are RES morphisms. We then need to prove that for any RES $\mathcal{E}_2$ such that there exist RES morphisms $f_0 : \mathcal{E}_2 \to \mathcal{E}_0$ and $f_1 : \mathcal{E}_2 \to \mathcal{E}_1$, there exists a unique RES morphism $f : \mathcal{E}_2 \to (\mathcal{E}_0 \times \mathcal{E}_1)$ such $\pi_0 \circ f = f_0$ and $\pi_1 \circ f = f_1$.

Since $E_0 \times_* E_1$, π_0, and π_1 comprise a product in the category of sets and partial functions, f must be unique.

We define f as $f(e) = (f_0(e), f_1(e))$ and prove that it is a morphism, meaning it satisfies the conditions of Definition 6.3:

1. Suppose $X \otimes Y \vdash_2 e^*$ and $f(e) \neq \perp$. If $f_0(e) \neq \perp$ then there exists a $Y_0 \subseteq E_0$ such that for any $e' \in E_2$, if $f_0(e') \in Y_0$ then $e' \in Y$, and $f_0(X) \otimes Y_0 \vdash_0 f_0(e)^*$, and if $f_1(e) \neq \perp$ then there exists a $Y_1 \subseteq E_1$ such that for any $e' \in E_2$, if $f_1(e') \in Y_1$ then $e' \in Y$, and $f_1(X) \otimes Y_1 \vdash_1 f(e)^*$. We define $Y' = \{e \mid \pi_0(e) \in Y_0 \text{ or } \pi_1(e) \in Y_1\}$, where $Y_i = \emptyset$ if $f_i(e) = \perp$, and it is obvious that $f(X) \otimes Y' \vdash f(e)^*$ and for any $e' \in E_2$, if $f(e') \in Y'$ then $e' \in Y$.
2. Suppose $X \in \mathsf{Con}_2$. Then $f_0(X) \in \mathsf{Con}_0$ and $f_1(X) \in \mathsf{Con}_1$, and there are no $e, e' \in X$ such that $e \neq e'$ and either $f_0(e) = f_0(e')$ or $f_1(e) = f_1(e')$, implying $f(X) \in \mathsf{Con}$.
3. Suppose $f(e) = f(e') \neq \perp$ and $e \neq e'$. Then $f_0(e) = f_0(e')$ and $f_1(e) = f_1(e')$. Since we cannot have $f_0(e) = f_1(e) = \perp$ without $f(e) = \perp$, and f_0 and f_1 are morphisms, there cannot exist $X \in \mathsf{Con}_2$ such that $e, e' \in X$. □

Example 6.9 shows the product of two RESs, represented by CSs in Fig. 7.

Example 6.9 *(RES product).* Consider the RESs corresponding to the CSs in Fig. 7, $\mathcal{E}_0 = (E_0, F_0, \mathsf{Con}_0, \vdash_0)$ and $\mathcal{E}_1 = (E_1, F_1, \mathsf{Con}_1, \vdash_1)$, where

$$E_0 = \{a, b\} \quad F_0 = E_0 \quad \mathsf{Con}_0 = 2^{E_0}$$
$$\emptyset \otimes \emptyset \vdash_0 a \quad \{a\} \otimes \emptyset \vdash_0 b \quad \{a, b\} \otimes \emptyset \vdash_0 \underline{b} \quad \{a\} \otimes \emptyset \vdash_0 \underline{a}$$

and

$E_1 = \{c\}$ $\quad F_1 = E_1$ $\quad \mathsf{Con}_1 = \{\emptyset, \{c\}\}$
$\emptyset \oslash \emptyset \vdash_1 c$ $\quad \{c\} \oslash \vdash_1 \underline{c}$

The product $\mathcal{E}_0 \times \mathcal{E}_1$ is $(E, F, \mathsf{Con}, \vdash)$ where

$E = F = \{(a, *), (b, *), (a, c), (b, c), (*, c)\}$
$\mathsf{Con} = \{\emptyset, \{(a, *)\}, \{(b, *)\}, \{(a, c)\}, \{(b, c)\}, \{(*, c)\}, \{(a, *), (b, *)\}$
$\{(a, *), (b, c)\}, \{(a, *), (*, c)\}, \{(a, c), (b, *)\}, \{(b, *), (*, c)\}, \{(a, *), (b, *), (*, c)\}\}$
with forwards events enabled by $\quad \emptyset \oslash \emptyset \vdash (a, *)$
$\{(a, *)\} \oslash \emptyset \vdash (b, *)$ $\quad \{(a, c)\} \oslash \emptyset \vdash (b, *)$ $\quad \emptyset \oslash \emptyset \vdash (a, c)$
$\{(a, *)\} \oslash \emptyset \vdash (b, c)$ $\quad \{(a, c)\} \oslash \emptyset \vdash (b, c)$, $\quad \emptyset \oslash \emptyset \vdash (*, c)$
and backwards events enabled by $\quad \{(a, *)\} \oslash \emptyset \vdash \underline{(a, *)}$
$\{(b, *), (a, *)\} \oslash \emptyset \vdash \underline{(b, *)}$ $\quad \{(b, *), (a, c)\} \oslash \emptyset \vdash \underline{(b, *)}$ $\quad \{(a, c)\} \oslash \emptyset \vdash \underline{(a, c)}$
$\{(b, c), (a, *)\} \oslash \emptyset \vdash \underline{(b, c)}$ $\quad \{(*, c)\} \oslash \emptyset \vdash \underline{(*, c)}$

We define a functor C_r from **RES** to **CS** (Definition 6.10). We remark that [21] described single event transitions of RESs and [22] discussed extending this to multiple events. We define configurations according with the first condition of Definition 2.23, disregarding the second condition, as we do not want to restrict ourselves to forwards-reachable configurations. For the transition, condition i. follows from the definition of a CS. The other two conditions require any event being added or removed in the transition to be enabled at any intermediate configuration during the transition, both when all the events being removed have been removed but nothing has been added, and also when all the events have been added, but none have been removed. They also require that the same enabling can be used in all intermediate configurations.

Definition 6.10 *(From **RES** to **CS**).* The functor $C_r : \mathbf{RES} \to \mathbf{CS}$ is defined as

1. $C_r((E, F, \mathsf{Con}, \vdash)) = (E, F, \mathsf{C}, \to)$, where
   (a) $X \in \mathsf{C}$ if for all $X' \subseteq_{\text{fin}} X$, $X' \in \mathsf{Con}$;
   (b) for $X, Y \in \mathsf{C}$, $X \xrightarrow{A \cup \underline{B}} Y$ if:
      i. $Y = (X \setminus B) \cup A$, $A \cap X = \emptyset$, $B \subseteq X$, and $X \cup A \in \mathsf{C}$;
      ii. for all $e \in A$, $X' \oslash Z \vdash e$ for some X', Z such that $X' \subseteq_{\text{fin}} X \setminus B$ and $Z \cap (X \cup A) = \emptyset$;
      iii. for all $e \in B$, $X' \oslash Z \vdash \underline{e}$ for some X', Z such that $X' \subseteq_{\text{fin}} X \setminus (B \setminus \{e\})$ and $Z \cap (X \cup A) = \emptyset$;
2. $C_r(f) = f$.

Applying this functor to an RES results in a *finitely enabled CS* (FCS). We call this 'finitely enabled' because the configurations are all determined by the set of finite configurations, and each transition from an infinite configuration X can be seen as enabled by a transition from a finite configuration X', such that every configuration between them can perform the same transition.

Definition 6.11. Let $\mathcal{C} = (E, F, \mathsf{C}, \to)$ be a CS. We say that $\mathcal{C}$ is *finitely enabled* if

1. C is downwards-closed;
2. $X \in \mathsf{C}$ if and only if for all $X' \subseteq_{\text{fin}} X$, $X' \in \mathsf{C}$;
3. whenever $X \xrightarrow{e^*}$, there exists a finite configuration $X' \subseteq_{\text{fin}} X$ such that $X'' \xrightarrow{e^*}$ for all X'' such that $X' \subseteq X'' \subseteq X$.

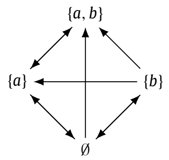
We call the category of FCSs and the CS morphisms between them **FCS**. Since this is a full subcategory of **CS**, Proposition 6.13 implies that we can define the product and coproduct in **FCS** in the same way as we do in **CS**.

Proposition 6.12. *C_r is a functor from **RES** to **FCS**.*

Proof. We first show that if $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ is an $(E, F, \mathsf{Con}, \vdash)$, then $C_r(\mathcal{E}) = (E, F, \mathsf{C}, \to)$ is a CS, as defined in Definition 3.1, meaning that for $X, Y \in \mathsf{C}$, $A \subseteq E$, and $B \subseteq F$, if $X \xrightarrow{A \cup \underline{B}} Y$, then:

1. $A \cap X = \emptyset$, $B \subseteq X \cap F$, and $Y = (X \setminus B) \cup A$ by definition.
2. For all $A' \subseteq A$ and $B' \subseteq B$, $(X \setminus B') \cup A = Z \in \mathsf{C}$ because $Z \subseteq X \cup A \in \mathsf{C}$. Moreover there exists a transition $X \xrightarrow{A' \cup \underline{B'}} Z$ because for $e \in A$, $X' \oslash Z' \vdash e$ for some $X' \subseteq_{\text{fin}} X \setminus B \subseteq X \setminus B'$, and $Z' \cap (X \cup A') \subseteq Z' \cap (X \cup A') = \emptyset$, and similarly for $b \in B'$. There exists a transition $Z \xrightarrow{(A \setminus A') \cup \underline{(B \setminus B')}} Y$ for similar reasons.

We then need to show that $C_r(\mathcal{E})$ is finitely enabled:



(a) FCS $\mathcal{C}$

$\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ where
$E = F = \{a, b\}$
$\mathsf{Con} = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$
$\emptyset \oslash \emptyset \vdash a$, $\emptyset \oslash \emptyset \vdash b$,
$\{b\} \oslash \emptyset \vdash \underline{b}$, $\{a\} \oslash \{b\} \vdash \underline{a}$

(b) RES $\mathcal{E}$

Fig. 15. An FCS and the corresponding RES such that $R(\mathcal{C}) = \mathcal{E}$ and $C_r(\mathcal{E}) = \mathcal{C}$.

1. Clearly C is downwards-closed, since Con is downwards-closed.
2. $X \in \mathsf{C}$ if and only if for all $X' \subseteq_{\text{fin}} X$, $X' \in \mathsf{Con}$, implying $X' \in \mathsf{C}$.
3. If $X \xrightarrow{e}$ then there exist X', Z such that $X' \oslash Z \vdash e$, $X' \subseteq_{\text{fin}} X$, and $Z \cap (X \cup \{e\}) = \emptyset$. Hence $X'' \xrightarrow{e}$ whenever $X' \subseteq X'' \subseteq X$. Similarly for $X \xrightarrow{\underline{e}}$.

We then show that an RES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$ is also a CS morphism (Definition 3.3) $f : C_r(\mathcal{E}_0) \to C_r(\mathcal{E}_1)$.:

1. This is implied by item 2 of the definition of an RES morphism.
2. Suppose $X \xrightarrow{A \cup \underline{B}}_0 Y$. Then $f(X) \xrightarrow{f(A) \cup f(\underline{B})}_1 f(Y)$ because:
   (a) $f(Y) = (f(X) \setminus f(B)) \cup f(A)$ because $Y = (X \setminus B) \cup A$ and $B \subseteq X$.
   $f(A) \cap f(X) = \emptyset$ because $A \cap X = \emptyset$, implying that if $e_1 \in f(A) \cap f(X)$ then there must exist $E_0 \in A$ and $e'_0 \in X$ such that $e_1 = f(e_0) = f(e'_0)$. However, by definition of an RES morphism, this implies that no $X' \in \mathsf{Con}_0$ exists such that $e_0, e'_0 \in X'$, which cannot be true, since $X \cup A \in \mathsf{Con}_0$.
   $f(B) \subseteq f(X)$ because $B \subseteq X$.
   $f(X) \cup f(A) \in \mathsf{C}_1$ according to item 1 because $X \cup A \in \mathsf{C}_0$.
   (b) For all a in A, $X' \oslash Z \vdash a$ for some X', Z such that $X' \subseteq_{fin} X \setminus B$ and $Z \cap (X \cup A) = \emptyset$. If $X' \oslash Z \vdash a$ then there exists a Z' such that for all $e_0 \in E_0$, if $f(e_0) \in Z'$ then $e_0 \in Z$ and $f(X') \oslash Z' \vdash_1 f(e)^*$. Since $X \subseteq B$ and $X' \subseteq_{fin} X \setminus B$, $f(X') \subseteq_{fin} f(X) \setminus f(B)$, and since $e_0 \in Z$ for all $e_0 \in E_0$ such that $f(e_0) \in Z'$, we have $Z' \cap (f(X) \cup f(A)) \subseteq f(Z \cap (X \cup A)) = \emptyset$.
   (c) Similar to previous condition.
3. This is implied by item 3 of the definition of an RES morphism.

Having proved that C_r maps from **RES** to **FCS**, the proof of it being a functor is similar to previous functors. □

Proposition 6.13. *Given FCSs $\mathcal{C}_0$ and $\mathcal{C}_1$, both $\mathcal{C}_0 + \mathcal{C}_1$ and $\mathcal{C}_0 \times \mathcal{C}_1$ are FCSs.*

Proof. Straightforward. □

We describe a functor, R, from **FCS** to **RES** in Definition 6.14. An example of a mapping between an FCS and RES can be seen in Fig. 15. The functor R works by taking each transition and creating the minimum enablings required for C_r to create that transition (that is, if $X \xrightarrow{A \cup \underline{B}} Y$ then we need $X \setminus B \oslash E \setminus (X \cup A) \vdash e$ for each $e \in A$, and similarly for $\underline{e} \in \underline{B}$) and adding the additional enablings required by the weakening condition of Definition 6.1 of RESs.

Definition 6.14 *(From **FCS** to **RES**).* The functor $R : \mathbf{FCS} \to \mathbf{RES}$ is defined as:

1. $R((E, F, \mathsf{C}, \to)) = (E, F, \mathsf{Con}, \vdash)$ where $X \in \mathsf{Con}$ if $X \subseteq_{\text{fin}} X'$ for some $X' \in \mathsf{C}$ and $X \oslash Y \vdash e^*$ if:
   (a) $X \cap Y = \emptyset$;
   (b) if $e^* = \underline{e}$ then $e \in X$;
   (c) there exists an $X' \subseteq E$ such that $X' \xrightarrow{A \cup \underline{B}}$ for some A and B such that $e^* \in A \cup \underline{B}$, $X' \setminus B \subseteq X$, and $Y = E \setminus (X' \cup A)$;
2. $R(f) = f$.

Proposition 6.15. *R : **FCS** $\to$ **RES** is a functor.*

Proof. We first show that given an FCS $\mathcal{C}$, $R(\mathcal{C})$ is an RES according to Definition 6.1:

1. Suppose $X \oslash Y \vdash e^*$. Then $X \cap Y = \emptyset$, and $Y = E \setminus (X' \cup A)$ for some X' and A such that $e^* \in A \cup \underline{X'}$. Therefore $(X \cup \{e\}) \cap Y = \emptyset$.
2. Suppose $X \oslash Y \vdash \underline{e}$, then clearly $e \in X$.

3. Suppose $X \oslash Y \vdash e^*$, $X \subseteq X' \in \mathsf{Con}$, and $X' \cap Y = \emptyset$. Then:
   (a) $X' \cap Y = \emptyset$;
   (b) if $e^* = \underline{e}$ then $e \in X \subseteq X'$;
   (c) there exists $X'' \subseteq E$ such that $X'' \xrightarrow{A\cup\underline{B}}$ for some A and B such that $e^* \in A \cup \underline{B}$, $X'' \setminus B \subseteq X \subseteq X'$ and $Y = E \setminus (X'' \cup A)$.

We then show that given a CS morphism $f : \mathcal{C}_0 \to \mathcal{C}_1$, we have that $f : R(\mathcal{C}_0) \to R(\mathcal{C}_1)$ is an RES morphism, satisfying the conditions of Definition 6.3:

1. Suppose $f(e) \neq \bot$ and $X \oslash Y \vdash_0 e^*$. Then there exists an $X' \subseteq E$ such that $X' \xrightarrow{A\cup\underline{B}}_0$ for some A and B such that $e^* \in A \cup \underline{B}$, $X' \setminus B \subseteq X$, and $Y = E \setminus (X' \cup A)$.
   Then $f(X) \oslash (E_1 \setminus (f(X') \cup f(A))) \vdash_1 f(e)^*$ because:
   (a) $X \cap Y = \emptyset$ and $f(Y) = f(E \setminus (X' \cup A)) \subseteq E_1 \setminus (f(X') \cup f(A))$, implying $f(X) \cap f(Y) = \emptyset$;
   (b) if $f(e)^* = \underline{f(e)}$ then $e \in X$, implying $f(e) \in f(X)$;
   (c) $f(X') \xrightarrow{f(A)\cup\underline{f(B)}}_1$, $f(e)^* \in f(A) \cup \underline{f(B)}$, and $f(X') \setminus f(B) \subseteq f(X)$.
   And for all $e \in E_0$, if $f(e_0) \in (E_1 \setminus (f(X') \cup f(A))$, then clearly $e_0 \in E \setminus (X' \cup A)$.
2. Suppose $X \in \mathsf{Con}_0$. Then there exists $X'_0 \in \mathsf{C}_0$ such that $X \subseteq_{\text{fin}} X'_0$. As $f(X'_0) \in \mathsf{C}_1$, and $f(X) \subseteq_{\text{fin}} f(X'_0)$, clearly $f(X) \in \mathsf{Con}_1$.
3. Suppose $f(e) = f(e')$ and $e \neq e'$. Then for all $X \in \mathsf{C}_0$, $\{e, e'\} \nsubseteq X$. Therefore for all $X' \in \mathsf{Con}_0$, $\{e, e'\} \nsubseteq X'$.

Having shown that R is a mapping from **CS** to **RES**, the rest of the proof is similar to previous functors. □

As Theorem 6.16 states, C_r and R are inverses of each other for CSs for which the set of configurations is downwards-closed and derived from the set of finite configurations, and no transitions starting from infinite configurations, which are not part of a larger transition going through a finite configuration. In fact, if $\mathcal{C}$ is an FCS, then $C_r(R(\mathcal{C}))$ will always be such an FCS.

Theorem 6.16. *Given an FCS $\mathcal{C} = (E, F, \mathsf{C}, \to)$, we have $C_r(R(\mathcal{C})) = \mathcal{C}$ if whenever $X \xrightarrow{A\cup\underline{B}}$, there exists X' such that $X' \xrightarrow{A'\cup\underline{B'}}$, $X' \setminus B'$ is finite, $A \subseteq A'$, $X \cup A \subseteq X' \cup A'$ and $B \subseteq B'$.*

Proof. Let $R(\mathcal{C}) = (E, F, \mathsf{Con}, \vdash)$ and $C_r((E, F, \mathsf{Con}, \vdash)) = (E, F, \mathsf{C}', \to')$. Then Con is the finite subsets of the configurations of C, so since $X \in \mathsf{C}$ iff $\forall X' \subseteq_{\text{fin}} X.X' \in \mathsf{C}$, obviously $C' = \{C \mid \forall X \subseteq_{\text{fin}} C.X \in \mathsf{Con}\} = \mathsf{C}$. For any $X, Y, A \subseteq E$ and $B \subseteq F$, there exists a transition $X \xrightarrow{A\cup\underline{B}}{}' Y$ if and only if:

1. $Y = (X \setminus B) \cup A$, $A \cap X = \emptyset$, $B \subseteq X$, and $X \cup A \in \mathsf{C}'$;
2. for all e in A, $X' \oslash Z \vdash e$ for some X', Z such that $X' \subseteq_{\text{fin}} X \setminus B$ and $Z \cap (X \cup A) = \emptyset$;
3. for all $e \in B$, $X' \oslash Z \vdash \underline{e}$ for some X', Z such that $X' \subseteq_{\text{fin}} X \setminus (B \setminus \{e\})$ and $Z \cap (X \cup A) = \emptyset$.

Item 1 holds for transitions in any CS, including $\mathcal{C}$. For any $e^* \in A \cup \underline{B}$, if $X' \subseteq X$, $X \cap Z = \emptyset$, $X \in \mathsf{Con}$, and $X' \oslash Z \vdash e^*$, then there must exist $X'' \subseteq E$ such that $X'' \xrightarrow{A'\cup\underline{B'}}$, $e^* \in A' \cup \underline{B'}$, $X'' \setminus B' \subseteq X$, and $Z = E \setminus (X'' \cup A')$. Therefore $X \xrightarrow{A\cup\underline{B}}$.

Suppose $X \xrightarrow{A\cup\underline{B}} Y$. Then $X \setminus B \xrightarrow{A} Y$.

If $X \setminus B$ is finite, and therefore in Con, we have $X \setminus B \oslash E \setminus (A \cup X) \vdash e$ for all $e \in A$, and for all $e \in B$ we have $X \setminus (B \setminus \{e\}) \xrightarrow{A\cup\{\underline{e}\}} Y$, similarly giving us $X \setminus (B \setminus \{e\}) \oslash E \setminus (A \cup X) \vdash \underline{e}$. Therefore $X \xrightarrow{A\cup\underline{B}}{}' Y$.

If $X \setminus B$ is infinite, then there exists X', A', B' such that $X' \xrightarrow{A\cup\underline{B}}$, $X \setminus B$ is finite, $A \subseteq A'$, $X \cup A \subseteq X' \cup A'$, and $B \subseteq B'$. This implies that for all $e \in A \subseteq A'$, $X' \setminus B' \oslash E \setminus (A' \cup X') \vdash e$ and for all $e \in B \subseteq B'$, $X' \setminus (B' \setminus \{e\}) \oslash E \setminus (A' \cup X') \vdash \underline{e}$. Therefore $X \xrightarrow{A\cup\underline{B}}{}' Y$. □

We now show that R and C_r form an adjunction between **RES** and **FCS**.

Theorem 6.17. $R \dashv C_r$.

Proof. For any RES $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$, we define $\epsilon : R(C_r(\mathcal{E})) \to \mathcal{E}$ such that for all $e \in E$ $\epsilon(e) = e$, and prove that it is an RES morphism.

If $C_r(\mathcal{E}) = (E, F, \mathsf{C}, \to)$ and $R(C_r(\mathcal{E})) = (E, \mathsf{Con}', \vdash')$ then:

1. Suppose $X \oslash Y \vdash' e^*$. Then clearly there exists $X' \subseteq E$ such that $X' \xrightarrow{A \cup \underline{B}}$ for some A, B such that $e^* \in A \cup \underline{B}$, $X' \setminus B \subseteq X$ and $Y = E \setminus (X' \cup A)$. And since $X' \xrightarrow{A \cup \underline{B}}$, we get $X'' \oslash Y' \vdash e^*$ for some X'', Y' such that $X'' \subseteq X' \setminus (B \setminus \{e\})$ and $Z \cap (X' \cup A) = \emptyset$. Obviously $Y' \subseteq Y$, and since $X'' \subseteq X$ and $X \cap Y = \emptyset$ we get $X \oslash Y' \vdash e^*$.
2. Suppose $X \in \mathsf{Con}'$. Then $X \subseteq_{\text{fin}} X' \in \mathsf{C}$, $\mathsf{C} = \{X' \mid \forall X''' \subseteq_{\text{fin}} X'.X' \in \mathsf{Con}\}$, and Con is downwards closed, implying $f(X) \in \mathsf{Con}$.
3. No $e, e' \in E$ exist such that $\epsilon(e) = \epsilon(e')$ and $e \neq e'$.

Since $\epsilon(e) = e$, it is obviously a natural transformation.

We then show that given an FCS $\mathcal{C}_0 = (E_0, F_0, \mathsf{C}_0, \rightarrow_0)$, an RES $\mathcal{E}_1 = (E_1, F_1, \mathsf{Con}_1, \vdash_1)$, and an RES morphism $f : R(\mathcal{C}_0) \rightarrow \mathcal{E}_1$, it holds that $f : \mathcal{C}_0 \rightarrow R(\mathcal{E}_1)$ is an FCS morphism.

Suppose $R(\mathcal{C}_0) = (E_0, F_0, \mathsf{Con}_0, \vdash_0)$ and $C_r(\mathcal{E}_1) = (E_1, F_1, \mathsf{C}_1, \rightarrow_1)$. Then f satisfies the conditions of Definition 3.3 because:

1. Suppose $C_0 \in \mathsf{C}_0$. Then for all $X \subseteq_{\text{fin}} C_0$, $X \in \mathsf{Con}_0$. Therefore $f(X) \in \mathsf{Con}_1$ if $f(X) \subseteq_{\text{fin}} f(C_0)$, and consequently $f(C_0) \in \mathsf{C}_1$.
2. Suppose $X \xrightarrow{A \cup \underline{B}}_0 Y$. Then $X \setminus B \oslash E \setminus (X \cup A) \vdash_0 e$ for all $e \in A$, and $X \setminus (B \setminus \{e\}) \oslash E \setminus (X \cup A) \vdash_0 \underline{e}$ for all $e \in B$. This implies that $f(X) \setminus (f(B) \setminus \{f(e)\}) \oslash Y_1 \vdash_1 f(e)^*$ for some Y_1 such that for all $e_0 \in E$, if $f(e_0) \in Y_1$ then $e_0 \in E \setminus (X \cup A)$. Therefore $f(X) \xrightarrow{f(A) \cup \underline{f(B)}}_1 f(Y)$.
3. If $C_0 \in \mathsf{C}_0$ then for all $X \subseteq_{\text{fin}} C_0$, $X \in \mathsf{Con}_0$, and if $f(e_0) = f(e_0') \neq \bot$, then there exists no $X_0 \in \mathsf{Con}_0$ such that $e_0, e_0' \in X_0$. □

We also show in Proposition 6.18 that C_r preserves both coproducts and products up to isomorphism, the latter being implied by Theorem 6.17.

Proposition 6.18. *Given RESs $\mathcal{E}_0$ and $\mathcal{E}_1$, $C_r(\mathcal{E}_0) + C_r(\mathcal{E}_1) = C_r(\mathcal{E}_0 + \mathcal{E}_1)$ and $C_r(\mathcal{E}_0) \times C_r(\mathcal{E}_1) \cong C_r(\mathcal{E}_0 \times \mathcal{E}_1)$.*

Proof. Let $\mathcal{E}_0 = (E_0, F_0, \mathsf{Con}_0, \vdash_0)$, $\mathcal{E}_1 = (E_1, F_1, \mathsf{Con}_1, \vdash_1)$, $\mathcal{E}_0 + \mathcal{E}_1 = (E, F, \mathsf{Con}, \vdash)$, $C_r(\mathcal{E}_0) = (E_0, F_0, \mathsf{C}_0, \rightarrow_0)$, $C_1(\mathcal{E}_1) = (E_1, F_1, \mathsf{C}_1, \rightarrow_1)$, $C_r(\mathcal{E}_0 + \mathcal{E}_1) = (E, F, \mathsf{C}, \rightarrow)$, and $C_r(\mathcal{E}_0) + C_r(\mathcal{E}_1) = (E', F', \mathsf{C}', \rightarrow')$. Then $C_r(\mathcal{E}_0) + C_r(\mathcal{E}_1) = C_r(\mathcal{E}_0 + \mathcal{E}_1)$ because:

1. $C = \{X \mid \exists i \in \{0,1\}, X_i.X = \{i\} \times X_i \text{ and } \forall X_i' \subseteq_{\text{fin}} X_i.X_i' \in \mathsf{Con}_i\} = \{X \mid \forall X' \subseteq_{\text{fin}} X.\exists i \in \{0,1\}, X_i \in \mathsf{Con}_i.X' = \{i\} \times X_i\} = \mathsf{C}'$.
2. If $X \xrightarrow{A \cup \underline{B}} Y$ then $X \xrightarrow{A \cup \underline{B}}{}' Y$ because:
   (a) $Y = (X \setminus B) \cup A$, $A \cap X = \emptyset$, $B \subseteq X$, and $X \cup A$.
   (b) For all $(i, e) \in A$, there exists $X' \oslash Z \vdash_i e$ for some X', Z such that $\{i\} \times X' \subseteq_{\text{fin}} X \subseteq B$ and $\{i\} \times Z \cap (X \cup A) = \emptyset$.
   (c) For all $(i, e) \in B$, there exists $X' \oslash Z \vdash_i \underline{e}$ for some X', Z such that $\{i\} \times X' \subseteq_{\text{fin}} X \subseteq (B \subseteq (i, e))$ and $\{i\} \times Z \cap (X \cup A) = \emptyset$.
3. If $X \xrightarrow{A \cup \underline{B}}{}' Y$ then $X \xrightarrow{A \cup \underline{B}} Y$ for similar reasons.

$C_r(\mathcal{E}_0) \times C_r(\mathcal{E}_1) \cong C_r(\mathcal{E}_0 \times \mathcal{E}_1)$ is implied by Theorem 6.17, since right adjoints preserve products up to isomorphism. □

Just as we had functors between **PES** and **RPES** and **AES** and **RAES**, we have a functor from **ES** to **RES**, extending the mapping from [21], described in Definition 6.19.

Definition 6.19 *(From **ES** to **RES**).* The functor $P_r : \mathbf{ES} \rightarrow \mathbf{RES}$ is defined as

1. $P_r((E, \mathsf{Con}, \vdash)) = (E, \emptyset, \mathsf{Con}, \vdash')$ where $X \oslash \emptyset \vdash' e$ if $X \vdash e$;
2. $P_r(f) = f$.

Proposition 6.20. *P_r is a functor from **ES** to **RES**.*

Proof. It was shown in [21] that for an ES $\mathcal{E}$, $P_r(\mathcal{E})$ is an RES.

If $f : \mathcal{E}_0 \rightarrow \mathcal{E}_1$ is an ES morphism, then $f : P_r(\mathcal{E}_0) \rightarrow P_r(\mathcal{E}_1)$ satisfies the conditions of Definition 6.3:

1. When $Y = Y_1 = \emptyset$, this is analogous to condition 1 of ES morphisms.
2. This is condition 2 of ES morphisms.
3. This is condition 3 of ES morphisms.

Once established that P_r is a mapping, the rest of the proof is similar to previous functors. □

We also define a mapping from RAESs to RESs in Definition 6.21. The possibility of doing so was first discussed in [22], though a mapping was not defined. The only time we have $C_{ra}(\mathcal{E}) = C_r(P_{ar}(\mathcal{E}))$ is when any set of events of the RAES which contains an infinite chain $e_0 \lhd e_1 \lhd e_2 \lhd \ldots$, also contains a finite $\lhd$-cycle. This is because, while RAES configurations require $\lhd$ to be well-founded, the set Con of the corresponding RES can only contain finite sets, such that all finite subsets of a configuration of the RES must be in Con. Obviously this does not allow us to define an RES with the same configurations as an RAES which has an infinite chain $e_0 \lhd e_1 \lhd e_2 \lhd \ldots$ without any finite $\lhd$-cycles. We do however still get an RES, and the extra configurations will never be reachable in a finite number of transitions. It is worth noting that, while we cannot map all RAESs to a corresponding RES, thanks to the preventing set, we can get much closer than in the forward-only setting. Recall from Remark 2.27 that such a mapping did not exist from **AES** to **ES**, due to ESs not having a notion of prevention. Also note that, since in RAESs generated by RPESs $a \rhd b$ if and only if $b \rhd a$, we have $C_{rp} = C_r \circ P_{ar} \circ A_r$. The mapping P_{ar} from **RAES** to **RES** also acts as a functor.

Definition 6.21 *(From **RAES** to **RES**).* The mapping $P_{ar} : \textbf{RAES} \to \textbf{RES}$ is defined as

1. $P_{ar}((E, F, \prec, \lhd)) = (E, F, \mathsf{Con}, \vdash)$ where:
   $\mathsf{Con} = \{X \subseteq_{\text{fin}} E \mid \lhd \text{ is well-founded on } X\}$:
   $X \oslash Y \vdash e$ if $\{e' \mid e' \prec e\} \subseteq X \in \mathsf{Con}$, $Y = \{e' \mid e' \rhd e\}$, $X \cap Y = \emptyset$, and $e \in E$:
   $X \oslash Y \vdash \underline{e}$ if $\{e' \mid e' \prec \underline{e}\} \subseteq X \in \mathsf{Con}$, $Y = \{e' \mid e' \rhd \underline{e}\}$, $X \cap Y = \emptyset$, and $e \in F$.
2. $P_{ar}(f) = f$.

Proposition 6.22. *P_{ar} is a functor from **RAES** to **RES**.*

Proof. We first show that given an RAES $\mathcal{E} = (E, F, \prec, \lhd)$, $P_{ar}(\mathcal{E})$ is an RES, satisfying the conditions of Definition 6.1:

1. Suppose $X \oslash Y \vdash e^*$. Then $(X \cup \{e\}) \cap Y = \emptyset$ because by definition of P_{ar}, $X \cap Y = \emptyset$ and by definition of an RAES, since $\lhd$ is irreflexive and $e \prec \underline{e}$ one cannot have $e \rhd e^*$.
2. Suppose $X \oslash Y \vdash \underline{e}$. Then $e \in X$ because $X = \{e' \in E \mid e' \prec \underline{e}\}$ and by definition $e \prec \underline{e}$.
3. Suppose $X \oslash Y \vdash e^*$, $X \subseteq X' \in \mathsf{Con}$, and $X' \cap Y = \emptyset$. Then it is clear from the definition of $\vdash$, that $X' \oslash Y \vdash e^*$.

We then show that given an RAES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$, $f : P_{ar}(\mathcal{E}_0) \to P_{ar}(\mathcal{E}_1)$ is an RES morphism, satisfying the conditions of Definition 6.3:

1. Suppose $X \oslash Y \vdash_0 e$. Then $\{e' \in E_0 \mid e' \prec_0 e\} \subseteq X \in \mathsf{Con}_0$, $Y = \{e' \in E_0 \mid e' \rhd_0 e\}$, and $X \cap Y = \emptyset$. Since $X \in \mathsf{Con}_0$, we have $f(X) \in \mathsf{Con}_1$. By definition of P_{pr}, $\{e' \in E_0 \mid e' \prec_0 e\} \subseteq X$, and by definition of an RAES morphism, $\{e_1 \mid e_1 \prec_1 f(e)\} \subseteq \{f(e') \mid e' <_0 e\}$, implying $\{e_1 \mid e_1 \prec_1 f(e)\} \subseteq f(X)$. And since $f(e') \rhd_1 f(e)$ implies $e' \rhd_0 e$, we get that for all $e' \in E_0$ such that $f(e') \in \{e' \mid e' \rhd_1 f(e)\}$, $e' \in Y$. If $e_1 \in f(X) \cap \{e' \mid e' \rhd_1 \underline{f(e)}\}$, then there must exist an $e_0 \in X$ such that $f(e_0) = e_1$, and therefore $e_0 \rhd_0 f(e)$, implying $e_0 \in X \cap Y$, which is not possible. Since $f(e) \in f(F_0) \subseteq F_1$, we can conclude that $f(X) \oslash \{e' \mid e' \rhd_1 f(e)\} \vdash_1 f(e)$.
   Suppose $X \oslash Y \vdash_0 \underline{e}$. Then the argument is similar.
2. Suppose $X_0 \in \mathsf{Con}_0$. Then $\lhd_0$ is well-founded on X_0, and for all $e, e' \in X_0$, if $f(e) \rhd_1 f(e')$, then $e \rhd_0 e'$. Hence $\lhd_1$ is also well-founded on $f(X_0)$, and therefore $f(X_0) \in \mathsf{Con}_1$.
3. Suppose $f(e) = f(e') \neq \perp$ and $e \neq e'$. Then $e \#_0 e'$ and $\mathsf{Con}_0 = \{Y \subseteq E_0 \mid \lhd_0 \text{ is well-founded on } Y\}$. Therefore $X \in \mathsf{Con}_0$ implies $\{e, e'\} \nsubseteq X$.

Having proved that P_{ar} is a mapping from **RAES** to **RES**, the rest of the proof is similar to previous functors. □

Theorem 6.23. *Given an RAES $\mathcal{E} = (E, F, \prec, \lhd)$, such that for any $X \subseteq E$, if $\lhd$ is not well-founded on X then there exists a finite $\lhd$-cycle in X, $C_r(P_{ar}(\mathcal{E})) = C_{ar}(\mathcal{E})$.*

Proof. Let $C_{ra}(\mathcal{E}) = (E, F, \mathsf{C}, \rightarrow)$, $P_{ar}(\mathcal{E}) = (E, F, \mathsf{Con}, \vdash)$, and $C_r(P_{ar}(\mathcal{E})) = (E, F, \mathsf{C}', \rightarrow')$.

We first prove that $\mathsf{C} = \mathsf{C}'$. Suppose $X \in \mathsf{C}$. Then $\lhd$ is well-founded on X, implying for all $X' \subseteq_{\text{fin}} X$, $\lhd$ is well-founded on X', and therefore $X' \in \mathsf{Con}$, implying $X \in \mathsf{C}'$.

Suppose $X' \in \mathsf{C}'$. This implies that for all $X \subseteq_{\text{fin}} X'$, $X \in \mathsf{Con}$ and therefore $\lhd$ is well-founded on X. Hence there cannot exist a finite $\lhd$-cycle in X', and therefore $\lhd$ is well-founded on X', implying $X' \in \mathsf{C}$.

It remains to show that $\rightarrow = \rightarrow'$. Suppose $X \xrightarrow{A \cup \underline{B}} Y$. Then $X, Y \in \mathsf{C}$, $A \subseteq E$, $B \subseteq F$, $Y = (X \setminus B) \cup A$, $A \cap X = \emptyset$, and $B \subseteq X$, so we need to prove that for all $a \in A$, $X' \oslash Z \vdash a$ for some $X' \subseteq X \setminus B$, Z, such that $Z \cap (A \cup X) = \emptyset$. Clearly this holds for $X' = \{e \mid e \prec a\}$ and $Z = \{e \mid e \rhd a\}$. Similarly for all $b \in B$, we have that $X' \oslash Z \vdash \underline{b}$ for some $X' \subseteq X \setminus (B \setminus \{b\})$, Z, such that $Z \cap (A \cup X) = \emptyset$.

Suppose $X \xrightarrow{A \cup \underline{B}}{}' Y$. Then $X, Y \in \mathsf{C}$, $A \subseteq E$, $B \subseteq F$, $Y = (X \setminus B) \cup A$, $A \cap X = \emptyset$, and $B \subseteq X$. Additionally for all $e \in A$, if $e' \prec e$ then whenever $X' \oslash Z \vdash e$, $e' \in X'$, and if $e' \triangleright e$ then whenever $X' \oslash Z \vdash e$, $e' \in Z$. Also for every $e \in B$, if $e' \prec \underline{e}$ then whenever $X' \oslash Z \vdash \underline{e}$, we have $e' \in X'$, and if $e' \triangleright \underline{e}$ then whenever $X' \oslash Z \vdash \underline{e}$, we have $e' \in Z$. □

7. Stable reversible event structures and configuration systems

Similarly to the stable event structures (Definition 2.25), we define the *stable reversible event structures* (SRES) in Definition 7.1, and create the category **SRES** consisting of SRESs and the RES morphisms between them. SRESs and SESs are defined identically, with the exception that in an SRES the preventing sets are included as well, and treated in much the same way as the enabling sets. Fig. 15b showed an SRES. Though we needed a non-stable RES to model the process described in Example 6.2, SRESs are still useful for other cases, as illustrated by Example 7.2. We also define a stable CS (SCS), and a new functor from SCSs to SRESs called R' (Definition 7.6). Functors R' and C_r act as inverses on SCSs (Proposition 7.13), and on SRESs in normal form (Definition 7.17).

Like in an SES, an event in a configuration of an SRES will have a single unique possible cause as long as the configuration has been reached by only going forwards. Also like the SES for the PES, mapping an RPES or RAES into an RES always results in an SRES, as stated in Proposition 7.3. In addition, the subcategory is closed under RES coproduct and product. Since **SRES** consisting of SRESs and the RES morphisms between them is a full subcategory, we can then construct SRES products and coproducts in the same manner as we do RES products and coproducts.

Definition 7.1 *(Stable RES).* A *stable* reversible event structure (SRES) is an RES $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ such that for all $e^* \in E \cup \underline{F}$, if $X \oslash Y \vdash e^*$, $X' \oslash Y' \vdash e^*$, and $X \cup X' + e^* \in \mathsf{Con}$, then $X \cap X' \oslash Y \cap Y' \vdash e^*$.

Example 7.2. Recall Example 1.4, where we wished to model a computation $e_0, e_1, e_2, \ldots$ which can be forced to reverse if it encounters one of a number of different errors $\{\mathsf{error}_0, \mathsf{error}_1, \ldots, \mathsf{error}_n\}$. We do this with an RES with events $\{e_0, e_1, \ldots\} \cup \{\mathsf{error}_0, \mathsf{error}_1, \ldots, \mathsf{error}_n\}$, consistent sets $\mathsf{Con} = \{X \mid \nexists i, j.\{\mathsf{error}_i, \mathsf{error}_j\} \subseteq X\}$, and the enablings:

$\{e_k \mid k < i\} \oslash \{\mathsf{error}_j \mid 0 \leq j \leq n\} \vdash e_i$ for all e_i,

$\emptyset \oslash \emptyset \vdash \mathsf{error}_j$ and $\{\mathsf{error}_j\} \oslash \{e_0\} \vdash \underline{\mathsf{error}_j}$ for all error_j,

and $\{\mathsf{error}_j\} \oslash \{e_k \mid k > i\} \vdash \underline{e_i}$ for all combinations of i and j.

This RES will be stable.

Proposition 7.3. *If $\mathcal{E} = (E, F, \prec, \lhd)$ is an RAES then $P_{ar}(\mathcal{E})$ is an SRES.*

Proof. Recall from Proposition 6.22, that $P_{ar}(\mathcal{E}) = (E, F, \mathsf{Con}, \vdash)$ is an RES.

We need to show that for all $e^* \in E \cup \underline{F}$, if $X \oslash Y \vdash e^*$, $X' \oslash Y' \vdash e^*$, and $X \cup X' + e^* \in \mathsf{Con}$ then $X \cap X' \oslash Y \cap Y' \vdash e^*$.

Suppose $e^* = e$. Then $\{e' \in E \mid e' < \underline{e}\} \subseteq X$, $X' \in \mathsf{Con}$ and $Y = Y' = \{e' \in E \mid e' \triangleright \underline{e}\}$. Since $\lhd$ is well-founded on $X \cup X' + e^*$, X and X' are finite, and $e' \triangleright e$ implies $e' \notin X \cup X'$, we know that $\lhd$ is well-formed on $X \cap X'$. Hence $\{e' \in E \mid e' < e\} \subseteq X \cap X' \in \mathsf{Con}$, implying that $X \cap X' \oslash Y \vdash e$ as required.

Suppose $e^* = \underline{e}$. Then $\{e' \in E \mid e' < \underline{e}\} \subseteq X$, $X' \in \mathsf{Con}$ and $Y = Y' = \{e' \in E \mid e' \triangleright \underline{e}\}$ and the logic is similar to above. □

Proposition 7.4. *Given two SRESs $\mathcal{E}_0$ and $\mathcal{E}_1$, $\mathcal{E}_0 + \mathcal{E}_1$ is an SRES.*

Proof. Let $\mathcal{E}_0 = (E_0, F_0, \mathsf{Con}_0, \vdash_0)$, $\mathcal{E}_1 = (E_1, F_1, \mathsf{Con}_1, \vdash_1)$, and $\mathcal{E}_0 + \mathcal{E}_1 = (E, F, \mathsf{Con}, \vdash)$. Then suppose $X \oslash Y \vdash e^*$, $X' \oslash Y' \vdash e^*$, and $X \cup X' + e^* \in \mathsf{Con}$. Then $e^* = (j, e_j)^*$ and $\exists X_j, Y_j \in E_j$ such that $X_j \oslash Y_j \vdash e^*$, $i_j(X_j) = X$, $Y = i_j(Y_j) \cup (E \setminus i_j(E_j))$ and $\exists X'_j, Y'_j \in E_j$ such that $X'_j \oslash Y'_j \vdash e^*$, $i_j(X'_j) = X'$, $Y' = i_j(Y'_j) \cup (E \setminus i_j(E_j))$. Since $X \cup X' + e^* \in \mathsf{Con}$ implies $X_j \cup X'_j + e^*_j \in \mathsf{Con}_0$, we get $X_j \cap X'_j \oslash Y_j \vdash e^*_j$. Therefore $X \cap X' \oslash Y \vdash e^*$. □

Proposition 7.5. *Given two SRESs $\mathcal{E}_0$ and $\mathcal{E}_1$, $\mathcal{E}_0 \times \mathcal{E}_1$ is an SRES.*

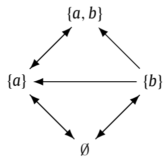
Proof. Let $\mathcal{E}_0 = (E_0, F_0, \mathsf{Con}_0, \vdash_0)$, $\mathcal{E}_1 = (E_1, F_1, \mathsf{Con}_1, \vdash_1)$, and $\mathcal{E}_0 \times \mathcal{E}_1 = (E, F, \mathsf{Con}, \vdash)$.

Then suppose $X \oslash Y \vdash e^*$, $X' \oslash Y' \vdash e^*$, and $X \cup X' + e^* \in \mathsf{Con}$. Then either $e^* = (e_0, *)^*$, $e^* = (*, e_1)^*$, or $e^* = (e_0, e_1)^*$

Suppose $e^* = (e_0, *)^*$. Then $\pi_0(X) \oslash Y_0 \vdash \pi_0(e)^*$, for some Y_0 such that $Y = \{e' \mid \pi_0(e') \in Y_0\}$, $\pi_0(X') \oslash Y'_0 \vdash \pi_0(e)^*$ for some Y'_0 such that $Y' = \{e' \mid \pi_0(e') \in Y'_0\}$, and $\pi_0(X \cup X' + e^*) \in \mathsf{Con}_0$. Hence $\pi_0(X) \cup \pi_0(X') + e^*_0 \in \mathsf{Con}_0$, and therefore $\pi_0(X) \cap \pi_0(X') \oslash \pi_0(Y) \vdash e^*_0$. Since if $e^* = \underline{e}$ then $e^* \in X \cap X'$, we therefore have $X \cap X' \oslash Y \vdash e^*$.

Suppose $e^* = (*, e_1)^*$. Then similar logic can be used.

Suppose $e^* = (e_0, e_1)^*$. Then $\pi_1(X) \oslash Y_1 \vdash \pi_1(e)^*$, $\pi_1(X') \oslash Y'_1 \vdash \pi_1(e)^*$, $\pi_0(X) \oslash Y_0 \vdash \pi_0(e)^*$, and $\pi_0(X') \oslash Y'_0 \vdash \pi_0(e)^*$, for some Y_0, Y'_0, Y_1, Y'_1 such that $Y = \{e' \mid \pi_0(e') \in Y_0$ or $\pi_1(e') \in Y_1\}$ and $Y' = \{e' \mid \pi_0(e') \in Y'_0$ or $\pi_1(e') \in Y'_1\}$, and $\pi_1(X \cup X' + e^*) \in \mathsf{Con}_1$ and $\pi_0(X \cup X' + e^*) \in \mathsf{Con}_0$, implying $\pi_0(X) \cup \pi_0(X') + e^*_0 \in \mathsf{Con}_0$ and $\pi_1(X) \cup \pi_1(X') + e^*_1 \in \mathsf{Con}_1$, and therefore $\pi_0(X) \cap \pi_0(X') \oslash Y_0 \vdash e^*_0$ and $\pi_1(X) \cap \pi_1(X') \oslash Y_1 \vdash e^*_1$, and, since if $e^* = \underline{e}$ then obviously $e^* \in X \cap X'$, $X \cap X' \oslash Y \vdash e^*$. □



(a) Non-stable CS $\mathcal{C}$

$\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ where
$E = F = \{a, b\}$
$\mathsf{Con} = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$
$\emptyset \oslash \emptyset \vdash a$,
$\{a\} \oslash \emptyset \vdash b$,
$\emptyset \oslash \{a\} \vdash b$,
$\{b\} \oslash \emptyset \vdash \underline{b}$, $\{a\} \oslash \{b\} \vdash \underline{a}$

(b) Non-stable RES $\mathcal{E}$

Fig. 16. Example of a non-stable CS and the corresponding RES.

Similarly, we also define a *stable configuration system* in Definition 7.6. Again we want events to have a unique cause in any configuration. Condition (a) states that if $X_1 \subseteq X_3$ and both X_1 and X_3 can do a transition, then so can every configuration in between them, since this means X_1 is unambiguously what enables this transition in X_3. Condition (b) is very similar to Definition 2.25, but referring to configurations rather than enabling sets. Condition (c) ensures that if $X_1 \cup X_2$ can do a transition, and X_1 and X_2 can each do the same transition individually, then that transition is caused by $X_1 \cap X_2$ in all those configurations. Finally, condition (d) states that if two transitions can both happen in any order between two states, then they are independent and can therefore happen simultaneously.

Definition 7.6 *(Stable CS).* A *stable* CS (SCS) is a CS $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$ such that:

1. C is downwards closed.
2. For $X_1, X_2, X_3 \in \mathsf{C}$, $A \subseteq E$, and $B \subseteq F$:
   (a) if $X_1 \subseteq X_2 \subseteq X_3$, and there exist transitions $X_1 \xrightarrow{A \cup \underline{B}}$ and $X_3 \xrightarrow{A \cup \underline{B}}$, then $X_2 \xrightarrow{A \cup \underline{B}}$;
   (b) if $((X_1 \cup X_2) \setminus B) \cup A \in \mathsf{C}$, and there exist transitions $X_1 \xrightarrow{A \cup \underline{B}}$ and $X_2 \xrightarrow{A \cup \underline{B}}$, then $X_1 \cap X_2 \xrightarrow{A \cup \underline{B}}$;
   (c) if $X_1 \cap X_2 \subseteq X_3 \subseteq X_1 \cup X_2$, $(X_3 \cup A) \setminus B \in \mathsf{C}$, and there exist transitions $X_1 \xrightarrow{A \cup \underline{B}}$, $X_2 \xrightarrow{A \cup \underline{B}}$, and $X_1 \cap X_2 \xrightarrow{A \cup \underline{B}}$, then $X_3 \xrightarrow{A \cup \underline{B}}$;
   (d) if $X_0, X_1, X_2, X_3 \in \mathsf{C}$, $A_0, A_1, B_0, B_1 \subseteq E$ and there exist transitions $X_0 \xrightarrow{A_0 \cup \underline{B_0}} X_1$, $X_0 \xrightarrow{A_1 \cup \underline{B_1}} X_2$, $X_1 \xrightarrow{A_1 \cup \underline{B_1}} X_3$, and $X_2 \xrightarrow{A_0 \cup \underline{B_0}} X_3$, then $X_0 \xrightarrow{A_0 \cup A_1 \cup \underline{B_0} \cup \underline{B_1}} X_3$.

Proposition 7.7. *Given an RAES $\mathcal{E} = (E, F, \prec, \lhd)$, $C_{ra}(\mathcal{E})$ is an SCS.*

Proof. Straightforward. □

Proposition 7.8. *Given SCSs $\mathcal{C}_0$ and $\mathcal{C}_1$, both $\mathcal{C}_0 + \mathcal{C}_1$ and $\mathcal{C}_0 \times \mathcal{C}_1$ are SCSs.*

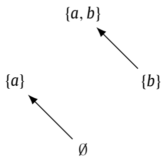
Proof. Straightforward. □

Fig. 15a shows a stable CS. One way to make it non-stable would be to remove the transition from $\emptyset$ to $\{a, b\}$, since that would violate item 2d, giving us the non-stable CS from Example 3.2, recalled in Fig. 16 with a corresponding RES added. This also makes the corresponding RES non-stable, as evidenced by the fact that $\{a\} \oslash \emptyset \vdash b$ and $\emptyset \oslash \{a\} \vdash b$, but not $\emptyset \oslash \emptyset \vdash b$.

We previously defined a functor R from **FCS** to **RES** (Definition 6.14), but some CSs can be represented by more than one RES. Consider, for example, the one seen in Fig. 17a, where the two transitions could be caused by the same or different enablings. The functor presented in Definition 6.14 gives us the RES in Fig. 17b. Note that while the FCS is stable, the RES generated by R is not. We would like a functor which maps SFCSs to SRESs, and we therefore define a new functor specifically for stable FCSs in Definition 7.9, which gives us the SRES in Fig. 17c. Functor R will always create an enabling for each transition (with the required subenablings caused by the weakening condition of Definition 6.1), while R' finds a maximum X and a minimum X' such that for $X' \subseteq X'' \subseteq X$, if $X'' \in \mathsf{C}$ and $X'' + e^* \in \mathsf{C}$ then $X'' \xrightarrow{\{e^*\}}$, and generates one enabling (with required subenablings) for all those configurations. For this reason R' does not preserve morphisms on non-stable FCSs, and we therefore only use it on stable ones. However, R' also gives us the ability to model transitions from infinite sets which are not a part of a larger transition from a finite set as described in Theorem 6.16.

Definition 7.9 *(From **SFCS** to **SRES**).* The functor $R' : \mathbf{SFCS} \rightarrow \mathbf{SRES}$ is defined as:

1. $R'((E, F, \mathsf{C}, \rightarrow)) = (E, F, \mathsf{Con}, \vdash)$ where $\mathsf{Con} = \{X \mid X \subseteq_{\mathrm{fin}} X' \in \mathsf{C}\}$ and for all X, X', e such that $X' \in \mathsf{C}$, $e \in E$, and $X \supseteq X'$ if:
   (a) if $X + e^* \in \mathsf{C}$ then $X \xrightarrow{\{e^*\}}$;



(a) CS $\mathcal{C}$

$R(\mathcal{C}) = (E, F, \mathsf{Con}, \vdash)$ where
$E = \{a, b\}$
$F = \emptyset$
$\mathsf{Con} = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$
$\emptyset \oslash \{b\} \vdash a$
$\{b\} \oslash \emptyset \vdash a$

(b) RES $R(\mathcal{C})$

$R'(\mathcal{C}) = (E, F, \mathsf{Con}, \vdash)$ where
$E = \{a, b\}$
$F = \emptyset$
$\mathsf{Con} = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$
$\emptyset \oslash \emptyset \vdash a$

(c) RES $R'(\mathcal{C})$

Fig. 17. Example of a CS and the two different corresponding RESs.

(b) if $X', X' + e^* \in \mathsf{C}$ then $X' \xrightarrow{\{e^*\}}$, and whenever $X' \subseteq X'' \subseteq X$ and $X'', X'' + e^* \in \mathsf{C}$, there exists a transition $X'' \xrightarrow{\{e^*\}}$;
(c) no $X'' \subsetneq X'$ exists such that if $X'', X'' + e^* \in \mathsf{C}$ then $X'' \xrightarrow{\{e^*\}}$, and whenever $X'' \subseteq X''' \subseteq X$ and $X''', X''' + e^* \in \mathsf{C}$ there exists a transition $X''', X''' + e^* \xrightarrow{\{e^*\}}$;
(d) no $X'' \supsetneq X$ exists such that if $X'', X'' + e^* \in \mathsf{C}$ then $X'' \xrightarrow{\{e^*\}}$, and whenever $X' \subseteq X''' \subseteq X''$ and $X''', X''' + e^* \in \mathsf{C}$ there exists a transition $X''' \xrightarrow{\{e^*\}}$;
(e) there exists $C \subseteq \mathsf{C}$, such that $\bigcup C = X$, and for all $X'' \in C$, $X'' \xrightarrow{\{e^*\}}$ and $X' \subseteq X''$;
then we get the following enablings:
(a) if $e^* = e$, then for all $X'' \in \mathsf{Con}$ such that $X' \subseteq X'' \subseteq X \cup \{e\}$, $X'' \oslash E \setminus (X \cup \{e\}) \vdash e$;
(b) if $e^* = \underline{e}$, then for all $X'' \in \mathsf{Con}$ such that $X' \subseteq X'' \subseteq X$, $X'' \oslash E \setminus X \vdash \underline{e}$.

2. $R'(f) = f$.

Proposition 7.10. *R' is a functor from **SFCS** to **SRES**.*

Proof. We first show that given an SFCS, $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$, $R'(\mathcal{C})$ is an RES, as described in Definition 6.1:

1. Suppose $X'' \oslash Y \vdash e^*$. Then clearly there exist X' and X such that $X' \subseteq X'' \subseteq X$, such that $Y = E \setminus X$, and therefore $X \cap Y = \emptyset$.
2. Suppose $X'' \oslash Y \vdash \underline{e}$. Then $e \in X''$, because there exists $X' \subseteq X''$ such that $X' \xrightarrow{\{\underline{e}\}}$.
3. Suppose $X'' \oslash Y \vdash e^*$, $X'' \subseteq X''' \in \mathsf{Con}$, and $X''' \cap Y = \emptyset$. Then, since $X'' \subseteq X''' \subseteq E \setminus Y$, clearly $X''' \oslash Y \vdash e^*$.

We then show that $R'(\mathcal{C})$ is stable:

Whenever $X_0 \oslash Y_0 \vdash e$, $X_1 \oslash Y_1 \vdash e$, and $X_0 \cup X_1 \cup \{e\} \in \mathsf{Con}$ we know that for all configurations $X'_0, X'_1 \in \mathsf{C}$, if $X_0 \subseteq X'_0 \subseteq E \setminus (Y_2 \cup \{e\})$ and $X_1 \subseteq X'_1 \subseteq E \setminus (Y_1 \cup \{e\})$ then $X'_0 \xrightarrow{\{e\}}$ and $X'_1 \xrightarrow{\{e\}}$. Hence $X_0 \cup X_1 \cup \{e\} \in \mathsf{C}$, implying $X_0 \cap X_1 \xrightarrow{\{e\}}$, and $X_0 \cap X_1 \subseteq X'_0 \cap X'_1 \subseteq X'_1$. Hence $X'_0 \cap X'_1 \xrightarrow{\{e\}}$, and therefore for all $X \in \mathsf{C}$, if $X_0 \cap X_1 \subseteq X \subseteq (E \setminus (Y_1 \cup \{e\})) \cup (E \setminus (Y_0 \cup \{e\}))$ then $X \xrightarrow{\{e\}}$. This gives us $X_0 \cap X_1 \oslash Y_0 \cap Y_1 \vdash e$.

If $X_0 \oslash Y_0 \vdash \underline{e}$, $X_1 \oslash Y_1 \vdash \underline{e}$, and $X_0 \cup X_1 \setminus \{e\} \in \mathsf{Con}$ the argument is similar.

We then show that given an SFCS morphism $f : \mathcal{C}_0 \to \mathcal{C}_1$, $f : R'(\mathcal{C}_0) \to R'(\mathcal{C}_1)$ is an RES morphism, satisfying the conditions of Definition 6.3:

1. Suppose $f(e) \neq \bot$ and $X'' \oslash Y \vdash_0 e^*$. Then there exists a minimum X' such that whenever $X' \subseteq X''' \subseteq E_0 \setminus Y$ for $X''' \in \mathsf{C}_0$, we have a transition $X''' \xrightarrow{\{e^*\}}$. Therefore $f(X''') \xrightarrow{\{f(e)^*\}}$. We also have $X' \cup \{e'\} \in \mathsf{C}_0$ for all $e' \in E_0 \setminus Y$, and C is downwards-closed, whenever $f(X') \subseteq X'''' \subseteq f(E_0 \setminus Y)$, $X'''' \xrightarrow{\{f(e)^*\}}$. This implies that there exists a Y_1 such that $f(X) \oslash Y_1 \vdash f(e)^*$, and if $f(e') \in Y_1$, then $e' \in Y$.
2. Suppose $X_0 \in \mathsf{Con}_0$. Then there exists $X'_0 \in \mathsf{C}_0$ such that $X_0 \subseteq_{\text{fin}} X'_0$. Clearly $f(X'_0) \in \mathsf{C}_1$ and $f(X_0) \subseteq_{\text{fin}} f(X'_0)$, implying $f(X_0) \in \mathsf{Con}_1$.
3. Suppose $f(e) = f(e')$ and $e \neq e'$. Then there does not exist $X \in \mathsf{C}_0$ such that $e, e' \in C$. Since for all $X_0 \in \mathsf{Con}_0$, there exists $X'_0 \in \mathsf{C}_0$ such that $X \subseteq_{\text{fin}} X'_0$, there also does not exist $X''_0 \in \mathsf{Con}_0$, such that $e, e' \in X''_0$. □

Proposition 7.11. *If $\mathcal{E}$ is an SRES then $C_r(\mathcal{E})$ is an SFCS.*

Proof. Let $\mathcal{E}$ be an SRES and recall from Proposition 6.12 that $C_r(\mathcal{E})$ is an FCS. Then we need to show that $C_r(\mathcal{E}) = (E, F, \mathsf{C}, \rightarrow)$ fulfils the conditions of Definition 7.6:

1. C is downwards-closed because $C_r(\mathcal{E})$ is an FCS.
2. For $X_1, X_2, X_3 \in \mathsf{C}$:

(a) This means that for $e^* \in A \cup \underline{B}$, there exist X_1', X_3', Y_1, Y_3 such that $X_1' \subseteq X_1 \setminus (B \setminus \{e\})$, $X_3' \subseteq X_3 \setminus (B \setminus \{e\})$, $X_1' \oslash Y_1 \vdash e^*$, $X_3' \oslash Y_3 \vdash e^*$, $Y_1 \cap (X_1 \cup A) = \emptyset$, and $Y_3 \cap (X_3 \cup A) = \emptyset$, and therefore, since $X_1' \cap X_3' \oslash Y_1 \cap Y_3 \vdash e^*$, we get $X_2 \xrightarrow{A\cup\underline{B}}$.

(b) This means that for $e^* \in A \cup \underline{B}$, there exist X_1', X_2', Y_1, Y_2 such that $X_1' \subseteq X_1 \setminus (B \setminus \{e\})$, $X_2' \subseteq X_2 \setminus (B \setminus \{e\})$, $X_1' \oslash Y_1 \vdash e^*$, $X_2' \oslash Y_2 \vdash e^*$, $Y_1 \cap (X_1 \cup A) = \emptyset$, and $Y_2 \cap (X_2 \cup A) = \emptyset$, and therefore, since $X_1' \cap X_2' \oslash Y_1 \cap Y_2 \vdash e^*$ and C is downwards-closed, we get $X_1 \cap X_2 \xrightarrow{A\cup\underline{B}}$.

(c) This means that for $e^* \in A_0 \cup \underline{B_0}$, there exist Y_0, $Y_1 \subseteq E \setminus X_1$, and $Y_2 \subseteq E \setminus X_2$, such that $X_1 \cap X_2 \oslash Y_0 \vdash e^*$, $X_1 \oslash Y_1 \vdash e^*$ and $X_2 \oslash Y_2 \vdash e^*$, and clearly $(X_1 \cap X_2) \cap X_1 \cap X_2 = X_1 \cap X_2$, and $Y_0 \cap Y_1 \cap Y_2 \subseteq E \setminus (X_1 \cup X_2)$, implying $X_1 \cap X_2 \oslash Y_0 \cap Y_1 \cap Y_2 \vdash e^*$. Hence $X_3 \xrightarrow{A\cup\underline{B}}$.

(d) This means that for all $e^* \in A_0 \cup \underline{B_0}$, there exist X_0', X_2', Y_0, Y_2 such that $X_0' \subseteq X_0 \setminus (B_0 \setminus \{e\})$, $X_2' \subseteq X_2 \setminus (B_0 \setminus \{e\})$, $X_0' \oslash Y_0 \vdash e^*$, $X_2' \oslash Y_2 \vdash e^*$, $Y_0 \cap (\overline{X_0} \cup A_0) = \emptyset$, and $Y_2 \cap (X_2 \cup A_0) = \emptyset$. Hence $X_0' \cap X_2' \oslash Y_0 \cap Y_2 \vdash e^*$ and $X_0 \cap X_2 \subseteq X_0 \setminus B_1$ and $Y_0 \cap Y_2 \subseteq Y_2$, and similarly for $e'^* \in A_1 \cup \underline{B_1}$. Therefore $X_0 \xrightarrow{A_0\cup A_1\cup\underline{B_0}\cup\underline{B_1}}$. □

Though R' and C_r are inverses on SFCSs, they are not inverses on all SRESs, only those whose enablings satisfy Definition 7.12 below. As Proposition 7.13 suggests, only one such SRES exists per SFCS, but there can be one or more non-stable RESs with *necessary* and *minimal* enablings which also generates the same SFCS, as Fig. 17 shows.

An enabling is considered necessary if it is not implied by other enablings. If for example we have an enabling $\{a\} \oslash \emptyset \vdash b$ then also having $\{a\} \oslash \{c\} \vdash b$ is unnecessary, as there cannot exist any configurations in which $\{a\} \oslash \{c\} \vdash b$ allows us to do b, but $\{a\} \oslash \emptyset \vdash b$ does not. Additionally, we consider an enabling necessary if it is needed because of a necessary enabling and the weakening condition of Definition 6.1.

An enabling is considered minimal if every event which cannot potentially be in the same configuration as both the enabling set and the event being enabled is in the preventing set. For example $\emptyset \oslash \emptyset \vdash b$ is not minimal if there do not exist any configurations of which $\{a, b\}$ is a subset, as the enabling $\emptyset \oslash \{a\} \vdash b$ would have had the same effect with a larger preventing set.

Definition 7.12. Given an RES, $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$, an enabling $X \oslash Y \vdash e^*$ is

1. *necessary* if there exists an $X' \subseteq X$ such that $X' \oslash Y \vdash e^*$ and there does not exist an enabling $X' \oslash Y' \vdash e^*$ for $Y' \subsetneq Y$;
2. *minimal* if for every e' such that $X \cup \{e, e'\} \notin \mathsf{C}$, $e' \in Y$.

If all the enablings in $\mathcal{E}$ are necessary and minimal, we say that $\mathcal{E}$ is in *normal form*, and we define the category **nfRES** (resp. **nfSRES**) of RESs (resp. SRESs) in normal form and the RES morphisms between them.

Proposition 7.13.

1. *If $\mathcal{E}$ is an SRES in normal form, then $R'(C_r(\mathcal{E})) = \mathcal{E}$.*
2. *If $\mathcal{C}$ is an SFCS, then $C_r(R'(\mathcal{C})) = \mathcal{C}$.*

Proof.

1. If $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$, $C_r(\mathcal{E}) = (E, F, \mathsf{C}, \rightarrow)$, and $R'(C_r(\mathcal{E})) = (E, F, \mathsf{Con}', \vdash')$, then we have $\mathsf{C} = \{C \mid X \subseteq_{\mathrm{fin}} C \Rightarrow X \in \mathsf{Con}\}$, and $\mathsf{Con}' = \{X \mid \exists X' \in \mathsf{C}. X \subseteq_{\mathrm{fin}} X'\}$, and $C_r(\mathcal{E})$ is an FCS, and therefore obviously $\mathsf{Con}' = \mathsf{Con}$.

   Suppose $X \oslash Y \vdash e$. Then for all $X' \in \mathsf{C}$, if $X' \cup \{e\} \in \mathsf{C}$ and $X \subseteq_{\mathrm{fin}} X' \subseteq E \setminus (Y \cup \{e\})$, then $X' \xrightarrow{\{e\}}$, which implies that there must exist some minimum $X'' \subseteq X$ and maximum $X''' \supset E \setminus (Y \cup \{e\})$, such that if $X'' \subseteq X'''' \subseteq X'''$, and $X'''', X'''' \cup \{e\} \in \mathsf{C}$, then $X'''' \xrightarrow{\{e\}}$. As $X \oslash Y \vdash e$ is a necessary enabling $E \setminus (X''' \cup \{e\}) = Y$, giving us $X \oslash Y \vdash' e$. Similar arguments apply if $X \oslash Y \vdash \underline{e}$.

   Suppose $X \oslash Y \vdash' e$. Then there exists an $X' \subseteq X$ such that $X' \xrightarrow{\{e\}}$, and if $X'', X'' + e \in \mathsf{C}$ and $X' \subseteq X'' \subseteq E \setminus (Y \cup \{e\})$, then $X \xrightarrow{\{e\}}$. As $\mathcal{E}$ is stable, this obviously requires $X' \oslash Y' \vdash e$ for some $Y' \supseteq Y$ such that if $X \cap Y' = \emptyset$ and $X \cap Y \neq \emptyset$ then $X \cup \{e\} \notin \mathsf{C}$. Since all enablings in $\mathcal{E}$ are minimal, it must be that $Y = Y'$, and therefore $X \oslash Y \vdash e$. Similar arguments apply if $X \oslash Y \vdash' \underline{e}$.
2. If $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$, $R'(\mathcal{C}) = (E, F, \mathsf{Con}, \vdash)$, and $C_r(R'(\mathcal{C})) = (E, F, \mathsf{C}', \rightarrow')$, then $\mathsf{C}' = \mathsf{C}$ for the same reasons as in Theorem 6.16.

   Suppose $X \xrightarrow{A\cup\underline{B}} Y$. Then for all $e^* \in A \cup \underline{B}$, whenever $X \subseteq X' \subseteq Y - e^*$, $X' \xrightarrow{\{e^*\}}$. Hence there exists $Y' \subseteq E \setminus (Y \cup B)$ such that $X \oslash Y' \vdash e^*$, and therefore $X \xrightarrow{A\cup\underline{B}}{}'$.

   Suppose $X \xrightarrow{A\cup\underline{B}}{}' Y$. Then for all $e^* \in A \cup \underline{B}$, there exists $X' \subseteq X$ and Y' such that $(X \cup A) \cap Y' = \emptyset$, and $X' \oslash Y' \vdash e^*$. This in turn requires $X'' \xrightarrow{\{e^*\}}$ whenever $X' \subseteq X'' \subseteq E \setminus (Y \cup \{e\})$. Since $\mathcal{C}$ is stable, this implies $X \xrightarrow{A\cup\underline{B}} Y$. □

Proposition 7.14. *If $\mathcal{C}$ is an SFCS, then $R'(\mathcal{C})$ is in normal form.*

Proof. Let $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$ be an SFCS and let $R'(\mathcal{C}) = \mathcal{E} = (E, F, \mathsf{Con}, \vdash)$. Consider an enabling $X \oslash Y \vdash e^*$.

We first show that it is necessary. If $e^* = e$ then there exists an $X_1 \subseteq X$, such that whenever $X_1 \subseteq X'_1 \subseteq E \setminus Y$ and $X'_1, X'_1 \cup \{e\} \in \mathsf{C}$ then $X'_1 \xrightarrow{\{e\}}$, no $X_2 \subsetneq X_1$ exists such that whenever $X_2 \subseteq X'_2 \subseteq E \setminus Y$ and $X'_2, X'_2 \cup \{e\} \in \mathsf{C}$ then $X'_2 \xrightarrow{\{e\}}$, and no $X_3 \supsetneq E \setminus (Y)$ exists such that whenever $X_1 \subseteq X'_3 \subseteq X_3$ and $X'_3, X'_3 \cup \{e\} \in \mathsf{C}$ then $X'_3 \xrightarrow{\{e\}}$. Hence that $X_1 \in \mathsf{Con}$ because $X_1 \subseteq X \in \mathsf{Con}$ and there cannot exist an enabling $X_1 \oslash Y'$ for $Y' \subsetneq Y$.

If $e^* = \underline{e}$ the proof is similar.

The enabling $X \oslash Y \vdash e^*$ is also minimal, since C is downwards-closed and there exists $C \subseteq \mathsf{C}$, such that $\bigcup C = E \setminus Y$, and for all $X' \in C$, $X' \xrightarrow{\{e^*\}}$ and $X \subseteq X'$. □

Corollary 7.15. ***nfSRES*** *and* ***SFCS*** *are isomorphic.*

Proof. Implied by Propositions 7.13 and 7.14. □

We showed in Propositions 7.4 and 7.5 that the coproduct and product of two stable RESs will always be stable. We now show that the coproduct and product of RESs in normal form will also be in normal form.

Proposition 7.16. *Given RESs $\mathcal{E}_0$ and $\mathcal{E}_1$ in normal form, both $\mathcal{E}_0 + \mathcal{E}_1$ and $\mathcal{E}_0 \times \mathcal{E}_1$ are in normal form.*

Proof. Straightforward. □

We define a way of mapping from RESs to normal form RESs in Definition 7.17, and in the following propositions show that the RES generated by this mapping will map to the same CS as the original. To get an RES in normal form, we first minimise the enablings by finding one not required by the weakening condition of Definition 6.1, $X' \oslash Y' \vdash e^*$, and then adding any event e', for which $X' \cup \{e, e'\} \notin \mathsf{Con}$ to Y', and considering this enabling as well as those implied by it and the weakening condition part of $\vdash_{\mathsf{min}}$. Out of these minimised enablings, we then take the necessary (Definition 7.12) ones, and those required by the necessary ones and the weakening condition.

Definition 7.17 *(RES normaliser N).* Given an RES $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$, we define the minimised enablings of $\mathcal{E}$, $\vdash_{\mathsf{min}}$ as $X \oslash Y \vdash_{\mathsf{min}} e^*$ if there exist $X', Y' \subseteq E$ such that:

1. $X \in \mathsf{Con}$ and $(X \cup \{e\}) \cap Y = \emptyset$;
2. $X' \oslash Y' \vdash e^*$;
3. $X' \subseteq X$;
4. no $X'' \subsetneq X'$ exists such that $X'' \oslash Y' \vdash e^*$;
5. $e' \in Y$ iff $e \in Y'$ or $X' \cup \{e, e'\} \notin \mathsf{Con}$.

We say that $X' \oslash Y' \vdash e^*$ *generated* $X \oslash Y \vdash_{\mathsf{min}} e^*$.

We then take the necessary minimised enablings, $X \oslash Y \vdash_{\mathsf{nec}} e^*$, such that

1. $X \oslash Y \vdash_{\mathsf{min}} e^*$;
2. there exists an $X_{\mathsf{min}} \subseteq X$ such that:
   (a) $X_{\mathsf{min}} \oslash Y \vdash_{\mathsf{min}} e^*$;
   (b) no $X' \subseteq X_{\mathsf{min}}$ and $Y' \subseteq Y$ exist such that $X' \oslash Y' \vdash_{\mathsf{min}} e^*$ and not both $X' = X_{\mathsf{min}}$ and $Y' = Y$.

We define the functor $N : \mathbf{RES} \rightarrow \mathbf{nfRES}$ as:

1. $N((E, F, \mathsf{Con}, \vdash)) = (E, F, \mathsf{Con}, \vdash_{\mathsf{nec}})$;
2. $N(f) = f$.

We show that N maps from RESs to RESs in normal form, which can be represented by the same CS (Propositions 7.18 and 7.19).

Proposition 7.18. *N is a functor from* ***RES*** *to* ***nfRES****.*

Proof. We first show that given an RES $\mathcal{E}$ it is the case that $N(\mathcal{E}) = (E, F, \mathsf{Con}, \vdash_{nec})$ is an RES. For every enabling $X \oslash Y \vdash_{nec} e^*$, by definition $(X \cup \{e\}) \cap Y = \emptyset$, and if $e^* = \underline{e}$ then, since $X' \oslash Y' \vdash \underline{e}$ for some $X' \subseteq X$, $e \in X$. If $X \subseteq X'' \in \mathsf{Con}$ and $X'' \cap Y = \emptyset$ then clearly $X' \oslash Y' \vdash e^*$ can be used to generate the enabling $X'' \oslash Y \vdash_{nec} e^*$ according to Definition 7.17.

Consider any enabling $X \oslash Y \vdash_{nec} e^*$. We show that it is both necessary and minimal.

We first show that it is necessary. From Definition 7.17 we know that there exists $X_{min} \subseteq X$ such that $X_{min} \oslash Y \vdash_{min} e^*$ and no $X' \subseteq X_{min}$ and $Y' \subseteq Y$ exist such that $X' \oslash Y' \vdash_{min} e^*$ and not both $X' = X_{min}$ and $Y' = Y$. Clearly this implies $X_{min} \oslash Y \vdash_{nec} e^*$. If for all $X' \subseteq X$ such that $X' \oslash Y \vdash_{nec} e^*$, there exists a $Y' \subsetneq Y$ such that $X' \oslash Y' \vdash_{nec} e^*$, then we have a contradiction when $X' = X_{min}$.

We then show that $X \oslash Y \vdash_{nec} e^*$ is minimal. If $X' \oslash Y' \vdash e^*$ generated $X \oslash Y \vdash_{min} e^*$, then $Y = Y' \cup \{e' \mid X' \cup \{e, e'\} \notin \mathsf{Con}\}$ and $X' \oslash Y \vdash_{min} e^*$. Clearly $X_{min} \subseteq X'$, and therefore whenever $X_{min} \cup \{e, e'\} \notin \mathsf{Con}$, $e' \in Y$.

We then need to show that given RESs $\mathcal{E}_0$ and $\mathcal{E}_1$ and RES morphism $f : \mathcal{E}_0 \to \mathcal{E}_1$, $f : N(\mathcal{E}_0) \to N(\mathcal{E}_1)$ is an RES morphism, satisfying the conditions of Definition 6.3:

1. Suppose $f(e) \neq \perp$ and $X \oslash Y \vdash_{0_{min}} e^*$. Then there exist X' and Y' such that $X' \oslash Y' \vdash e^*$, $X' \setminus X$, no $X'' \subsetneq X'$ exists such that $X'' \oslash Y \vdash e^*$, and $e' \in Y$ iff $e \in Y'$ or $X' \cup \{e, e'\} \notin \mathsf{Con}$. Therefore $f(X') \oslash Y_1' \oslash\vdash_1 f(e)^*$ for some Y_1' such that if $f(e') \in Y_1$ then $e' \in Y$. We then find a minimum $X'' \subseteq f(X')$ such that $X'' \oslash Y_1' \vdash_1 f(e)^*$, and we need to show $f(X) \oslash Y_1' \cup \{e' \mid X'' \cup \{f(e), e'\} \notin \mathsf{Con}_1\} \vdash_{1_{min}} f(e)^*$ and if $f(e') \in Y_1' \cup \{e' \mid X'' \cup \{f(e), e'\} \notin \mathsf{Con}_1\}$ then $e' \in Y$.
   We show the latter first. Suppose $f(e') \in Y_1'$. Then $e' \in Y' \subseteq Y$. Suppose $f(e') \in \{e' \mid X'' \cup \{f(e), e'\} \notin \mathsf{Con}_1\}$. Then since $X'' \subseteq f(X')$ we know there exists $X_0'' \subseteq X'$ such that $f(X_0'') = X''$ and $X'' \cup \{e, e'\} \notin \mathsf{Con}_0$. Since Con_0 is downwards-closed we then have $X' \cup \{e, e'\} \notin \mathsf{Con}_0$, and therefore $e' \in Y$.
   We then show $f(X) \oslash Y_1' \cup \{e' \mid X'' \cup \{f(e), e'\} \notin \mathsf{Con}_1\} \vdash_{1_{min}} f(e)^*$. Clearly $f(X) \in \mathsf{Con}_1$, and $(f(X) \cup \{f(e)\}) \cap (Y_1' \cup \{e' \mid X'' \cup \{f(e), e'\}) = \emptyset$ because, as shown above, if $f(e') \in Y_1' \cup \{e' \mid X'' \cup \{f(e), e'\}$ then $e' \in Y$, and therefore $(X \cup \{e\}) \cap Y = \emptyset$. Hence $X'' \oslash Y_1' \vdash_1 f(e)^*$ can generate $f(X) \oslash Y_1' \cup \{e' \mid X'' \cup \{f(e), e'\} \notin \mathsf{Con}_1\} \vdash_{1_{min}} f(e)^*$.
   Suppose $f(e) \neq \perp$ and $X \oslash Y \vdash_{0_{nec}} e^*$. Then $X \oslash Y \vdash_{0_{min}} e^*$ and there exists an $X_{min} \subseteq X$ such that $X_{min} \oslash Y \vdash_{0_{min}} e^*$ and no $X' \subseteq X_{min}$ and $Y' \subseteq Y$ exist such that $X' \oslash Y' \vdash_{0_{min}} e^*$ and not both $X_{min} = X'$ and $Y = Y'$. There therefore must exist a minimal Y_1 such that $f(X_{min}) \oslash Y_1 \vdash_{1_{min}} f(e)^*$ and if $f(e') \in Y_1$ then $e' \in Y$, and a minimal $X_{1_{min}} \subseteq f(X_{min})$ such that $X_{1_{min}} \oslash Y_1 \vdash_{1_{min}} f(e)^*$ and therefore $X_{1_{min}} \oslash Y_1 \vdash_{1_{nec}} f(e)^*$. Clearly $X_{1_{min}} \subseteq f(X)$, $f(X) \in \mathsf{Con}_1$, and $f(X) \cap Y_1 = \emptyset$. Hence, since $N(\mathcal{E}_1)$ is an RES, $f(X) \oslash Y_1 \vdash_{1_{nec}} f(e)^*$.
2. N does not change Con.
3. N does not change Con.

Once established that N is a mapping from **RES** to **nfRES**, the rest of the proof is similar to previous functors. □

Proposition 7.19. *Given an RES $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$, $C_r(\mathcal{E}) = C_r(N(\mathcal{E}))$.*

Proof. Suppose $C_r(\mathcal{E}) = (E, F, \mathsf{C}, \to)$, $N(\mathcal{E}) = (E, F, \mathsf{Con}, \vdash')$, and $C_r(N(\mathcal{E})) = (E, F, \mathsf{C}, \to')$. Then whenever $X \xrightarrow{A \cup \underline{B}} Y$, we have $Y = (X \setminus B) \cup A$, $A \cap X = \emptyset$, $B \subseteq X$, and $X \cup A \in \mathsf{C}$.

There exist X' and Z such that $X' \oslash Z \vdash e$, $X' \subseteq_{fin} X$, and $Z \cap (X \cup A) = \emptyset$. Therefore we get that $X' \oslash (Z \cup \{e' \mid X' \cup \{e, e'\} \notin \mathsf{Con}\}) \vdash_{min} e$ and therefore $X' \oslash Z'' \vdash_{nec} e$ for $Z'' \subseteq Z \cup \{e' \mid X' \cup \{e, e'\} \notin \mathsf{Con}\}$.

Since $X \cup A \in \mathsf{C}$, we have for all $X'' \subseteq_{fin} X \cup A$, $X'' \in \mathsf{Con}$, and therefore $(\{e' \mid X' \cup \{e, e'\} \notin \mathsf{Con}\}) \cap (X \cup A) = \emptyset$.

We use similar logic to show that for all $e \in B$, $X' \oslash Z \vdash' \underline{e}$ for some X' and Z such that $X' \subseteq_{fin} X \setminus (B \setminus \{e\})$ and $Z \cap (X \cup A) = \emptyset$.

If $X \xrightarrow{A \cup \underline{B}}{}' Y$ then $Y = (X \setminus B) \cup A$, $A \cap X = \emptyset$, $B \subseteq X$, and $X \cup A \in \mathsf{C}$.

There exist X' and Z such that $X' \oslash Z \vdash_{nec} e$, $X' \subseteq_{fin} X$, and $Z \cap (X \cup A) = \emptyset$. Hence there exist $X'' \subseteq X'$ and $Z' \subseteq Z$ such that $X'' \oslash Z' \vdash e$.

We use similar logic to show that for all $e \in B$, $X' \oslash Z \vdash \underline{e}$ for some X' and Z such that $X' \subseteq_{fin} X \setminus (B \setminus \{e\})$ and $Z \cap (X \cup A) = \emptyset$. □

Corollary 7.20. *If $\mathcal{E}$ is an SRES in normal form, then $N(\mathcal{E}) = \mathcal{E}$.*

Proof. Implied by Propositions 7.13, 7.18, and 7.19. □

Corollary 7.21. *If $\mathcal{E}$ is an SRES, then $N(\mathcal{E}) = R'(C_r(\mathcal{E}))$.*

Proof. Implied by Propositions 7.13 and 7.19. □

In some cases, a non-stable RES which maps to a stable CS can become stable when normalised, as in Example 7.22. In other cases, such as the one seen in Fig. 17b, a non-stable RES in normal form can map to a stable CS.

Example 7.22. Consider the RES $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$, where

$$E = \{a, b, c\}$$
$$F = \emptyset$$
$$\mathsf{Con} = \{\emptyset, \{a\}\}$$
$$\emptyset \oslash \{b\} \vdash a$$
$$\emptyset \oslash \{c\} \vdash a$$

This is clearly neither normal nor stable, but $N(\mathcal{E})$ instead only has the enabling $\emptyset \oslash \{b, c\} \vdash a$, which makes it both normal and stable.

We have now defined the categories and functors seen in Fig. 9.

8. Cause-respecting and causal RESs and CSs

We define cause-respecting and causal subcategories **crRES** and **CRES** of **RES** in Definitions 8.1 and 8.2. We show that the previously introduced functors preserve these properties between the categories. The definitions are based on the idea that if $\underline{e}$ is enabled in a configuration $X \cup \{e\}$, then e is enabled in X.

Definition 8.1 *(Cause-respecting RES).* An RES $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ is *cause-respecting* if for any $X \in \mathsf{Con}$, $Y \subseteq E$, and $e \in F$, if $X \oslash Y \vdash \underline{e}$, then $X \setminus \{e\} \oslash Y' \vdash e$ for some $Y' \subseteq Y$.

Definition 8.2 *(Causal RES).* An RES $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ is *causal* if, given $X \in \mathsf{Con}$, $Y \subseteq E$, and $e \in F$ such that $X \cup \{e\} \in \mathsf{Con}$, $X \oslash Y \vdash e$ iff $X \cup \{e\} \oslash Y \vdash \underline{e}$.

We furthermore define a cause-respecting and causal CS (crCS and CCS) in much the same way as a CRES in Definitions 8.3 and 8.4. The categories **crCS** and **CCS** consist of respectively crCSs and CCSs, and the CS morphisms between them. The definition of a causal CS is basically the looping condition from [8], as the square condition from [8] is already part of the definition of a CS.

Definition 8.3 *(Cause-respecting CS).* A CS $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$ is *cause-respecting* if for any $X \in \mathsf{C}$ and $B \subseteq F$, if $X \xrightarrow{A \cup \underline{B}} (X \cup A) \setminus B$, then $X \setminus B \xrightarrow{A \cup B} X \cup A$.

Definition 8.4 *(Causal configuration system).* A CS $\mathcal{E} = (E, F, \mathsf{C}, \rightarrow)$ is *causal* if for any configuration $X \in \mathsf{C}$, $A, B \subseteq F$, if $X \xrightarrow{A \cup \underline{B}} (X \cup A) \setminus B$, then $(X \cup A) \setminus B \xrightarrow{B \cup \underline{A}} X$.

We show that C_r (Definition 6.10), R (Definition 6.14), and R' (Definition 7.9) preserve the cause-respecting and causal properties.

Proposition 8.5.

1. *If $\mathcal{E}$ is a crRES, then $C_r(\mathcal{E})$ is a crFCS.*
2. *If $\mathcal{C}$ is a crFCS, then $R(\mathcal{E})$ is a crRES.*
3. *If $\mathcal{C}$ is a crSFCS, then $R'(\mathcal{E})$ is a crSRES.*

Proof.

1. Suppose $\mathcal{E} = (E, F, \mathsf{Con}, \vdash)$ and $C_r(\mathcal{E}) = (E, F, \mathsf{C}, \rightarrow)$. Then whenever $X \in \mathsf{C}$ is a configuration, $A \subseteq E$, $B \subseteq F$, and $X \xrightarrow{A \cup \underline{B}} (X \cup A) \setminus B$, we have that (1) for all $a \in A$, there exist $X' \subseteq_{\mathrm{fin}} X \setminus B$ and $Z \subseteq E \setminus (X \cup A)$ such that $X' \oslash Z \vdash a$ and (2) for all $b \in B$, there exist $X' \subseteq_{\mathrm{fin}} X \setminus (B \setminus \{b\})$ and $Z \subseteq E \setminus (X \cup A)$ such that $X' \oslash Z \vdash \underline{b}$. Hence $X' \setminus \{b\} \oslash Z \vdash b$, and we therefore have $X \setminus B \xrightarrow{A \cup B} X \cup A$.
2. Suppose $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$ and $R(\mathcal{C}) = (E, F, \mathsf{Con}, \vdash)$. Then whenever $X \oslash Y \vdash \underline{e}$, by definition of R, there exists $X' \in \mathsf{C}$, $A \subseteq E$, and $B \subseteq F$ such that $X' \xrightarrow{A \cup \underline{B}}$, $e \in B$, $X' \setminus B \subseteq X$, and $Y = E \setminus (X' \cup A)$. We therefore have $X' \setminus B \xrightarrow{A \cup B} X' \cup A$, and consequently $X \setminus \{e\} \oslash Y \vdash e$.
3. Suppose $\mathcal{C} = (E, F, \mathsf{C}, \rightarrow)$ and $R(\mathcal{C}) = (E, F, \mathsf{Con}, \vdash)$. Then whenever $X \oslash Y \vdash \underline{e}$, by definition of R' we have for all $X' \in \mathsf{C}$, if $X \subseteq X' \subseteq (E \setminus Y)$, $X' \xrightarrow{\underline{e}} X' \setminus \{e\}$, implying $X' \setminus \{e\} \xrightarrow{\{e\}} X'$. We therefore get $X \setminus \{e\} \oslash Y' \vdash e$ for some $Y' \subseteq Y$. □

Proposition 8.6.

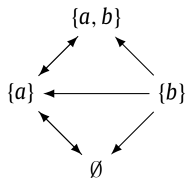
1. *If $\mathcal{E}$ is a CRES then $C_r(\mathcal{E})$ is a CFCS.*
2. *If $\mathcal{C}$ is a CFCS then $R(\mathcal{E})$ is a CRES.*
3. *If $\mathcal{C}$ is a CSFCS then $R'(\mathcal{C})$ is a CSRES.*

Proof.

1. Suppose $X \in \mathsf{C}$ is a configuration, $A, B \subseteq F$, and $X \xrightarrow{A \cup \underline{B}} (X \setminus B) \cup A$. Then $(X \setminus B) \cup A \xrightarrow{B \cup \underline{A}} X$ according to the definition of C_r (Definition 6.10) because:
   (a) Obviously $(X \setminus B) \cup A = (X \setminus B) \cup A$, $B \cap ((X \setminus B) \cup A) = \emptyset$, $A \subseteq ((X \setminus B) \cup A)$, and $((X \setminus B) \cup A) \cup B = X \cup A \in \mathsf{C}$.
   (b) For all $a \in A$, we have $X' \obslash Z \vdash e$ for some X', Z such that $X' \subseteq_{\mathrm{fin}} X \setminus B$ and $Z \cap (X \cup A) = \emptyset$. Therefore $X' \cup \{e\} \obslash Z \vdash \underline{e}$, and consequently $X' \cup \{e\} \subseteq_{\mathrm{fin}} ((X \setminus B) \cup A) \setminus (A \setminus e) = (X \cup \{e\}) \setminus B$ and $Z \cap (((X \setminus B) \cup A) \cup B) = \emptyset$.
   (c) For all $e \in B$, we have $X' \obslash Z \vdash \underline{e}$ for some X', Z such that $X' \subseteq_{\mathrm{fin}} X \setminus (B \setminus \{e\})$ and $Z \cap (X \cup A) = \emptyset$. Therefore $X' \setminus \{e\} \obslash Z \vdash e$, and consequently $X' \setminus \{e\} \subseteq_{\mathrm{fin}} ((X \setminus B) \cup A) \setminus A = X \setminus B$ and $Z \cap (((X \setminus B) \cup A) \cup B) = \emptyset$.
2. Suppose $X \obslash Y \vdash e$. Then there exists a transition $X' \xrightarrow{A \cup \underline{B}} E \setminus (Y \cup B)$, such that $X' \setminus B \subseteq X$ and $e \in A$. Hence $E \setminus (Y \cup B) \xrightarrow{B \cup \underline{A}} X'$. As $(E \setminus (Y \cup B)) \setminus A = (X' \cup A) \setminus B \subseteq X \setminus \{e\}$ and $Y = E \setminus ((E \setminus (Y \cup B)) \cup B)$, we get $X \cup \{e\} \obslash Y \vdash \underline{e}$.
   Similar arguments apply when $X \obslash Y \vdash \underline{e}$.
3. Suppose $X \obslash Y \vdash e$. Then Y is a maximum set of events such that for all $X' \in \mathsf{C}$ such that $X \subseteq X' \subseteq E \setminus Y$ and $X' \cup \{e\} \in \mathsf{C}$, there exists a transition $X' \xrightarrow{\{e\}}$. Therefore Y is a maximum set of events such that for all $X' \in \mathsf{C}$ such that $X \cup \{e\} \subseteq X' \subseteq E \setminus Y$ and $X' \setminus \{e\} \in \mathsf{C}$, there exists a transition $X' \xrightarrow{\{\underline{e}\}}$. Therefore we get $X \cup \{e\} \obslash Y \vdash \underline{e}$.
   Similar arguments apply when $X \obslash Y \vdash \underline{e}$. $\square$

The cause-respecting and causal RPESs and RAESs of Sections 4 and 5 cannot always be modelled by cause-respecting and causal CSs and RESs, as shown by Example 8.7. However, the reachable part of the CS is cause-respecting (resp. causal) and forwards-reachable, as proved in Theorem 4.54 and Propositions 3.37, 3.38, and 4.57 of [19]. We therefore define reachably cause-respecting and reachably causal CSs (rcrCS and rCCS) in Definition 8.8. We define categories **rcrCS** and **rCCS** of rcrCSs and rCCSs and the CS morphisms between them.

Example 8.7. Consider the RAES $\mathcal{E} = (E, F, \prec, \lhd)$ where $E = F = \{a, b\}$, $a \prec b$, $a \prec \underline{a}$, $b \prec \underline{b}$, and $b \rhd \underline{a}$. This is clearly causal, however when mapping to an RES, $P_{ar}(\mathcal{E}) = (E, F, \mathsf{Con}, \vdash)$, we get the enablings $\emptyset \obslash \emptyset \vdash a$, $\{a\} \obslash \emptyset \vdash b$, $\{b\} \obslash \emptyset \vdash \underline{b}$, and $\{a\} \obslash \{b\} \vdash \underline{a}$. Clearly this is not causal. This is also expressed in the CS in the following diagram:



Here b can be reversed from any configuration containing b, but a can only be reversed if b is not present. Conversely, a can be added to any configuration not containing a, but b can only be added if a is present. However, this gives us the unreachable configuration $\{b\}$, from which we have three transitions without corresponding reverse transitions.

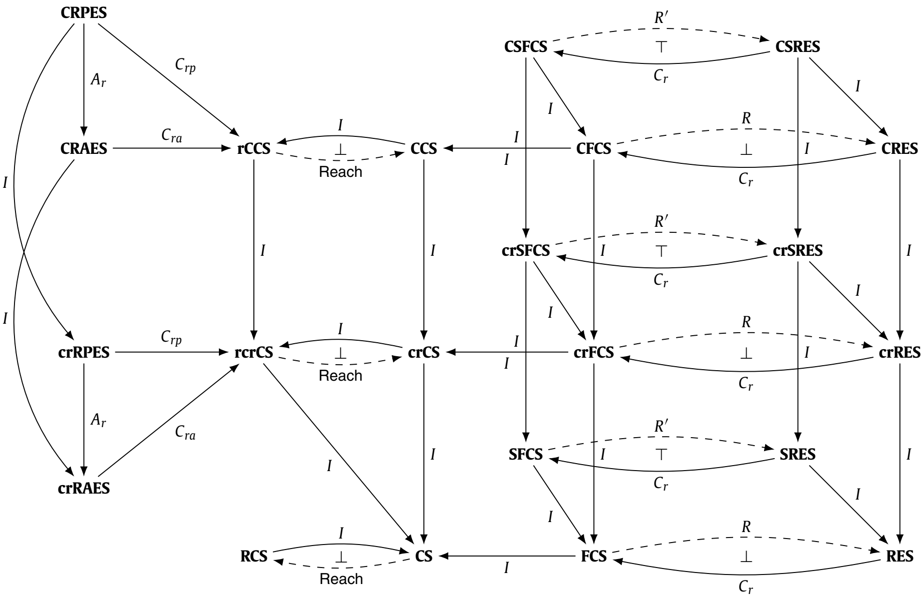
Definition 8.8 *(Reachably cause-respecting and causal CS).*

1. A CS $(E, F, \mathsf{C}, \rightarrow)$ is *reachably cause-respecting* if for any reachable configuration $X \in \mathsf{C}$, $A \subseteq E$, and $B \subseteq F$, if $X \xrightarrow{A \cup \underline{B}} (X \cup A) \setminus B$, then $X \setminus B \xrightarrow{A \cup \underline{B}} X \cup A$.
2. A CS $\mathcal{E} = (E, F, \mathsf{C}, \rightarrow)$ is *reachably causal* if for any reachable configuration $X \in \mathsf{C}$, $A, B \subseteq F$, if $X \xrightarrow{A \cup \underline{B}} (X \cup A) \setminus B$, then $(X \cup A) \setminus B \xrightarrow{B \cup \underline{A}} X$.

Clearly, $\mathcal{C}$ is reachably cause-respecting or causal if and only if $\mathsf{Reach}(\mathcal{C})$ is cause-respecting or causal.

Proposition 8.9.

1. *If $\mathcal{E}$ is a crRAES, then $C_{ra}(\mathcal{E})$ is an rcrCS.*
2. *If $\mathcal{E}$ is a CRAES, then $C_{ra}(\mathcal{E})$ is an rCCS.*



CS	Definition 3.1		**SRES**	Definition 7.1		Reach	Definition 3.17
RCS	Definition 3.16		**crRES**	Definition 8.1		C_{rp}	Definition 4.10
CRPES	Definition 4.18		**CRES**	Definition 8.2		C_{ra}	Definition 5.7
crRPES	Definition 4.18		**crCS**	Definition 8.3		A_r	Definition 5.15
CRAES	Definition 5.17		**CCS**	Definition 8.4		C_r	Definition 6.10
crRAES	Definition 5.17		**rCCS**	Definition 8.8		R	Definition 6.14
RES	Definition 6.1		**rcrCS**	Definition 8.8		R'	Definition 7.9
FCS	Definition 6.11						

Fig. 18. Categories of reversible event structures and their cause-respecting and causal subcategories and functors between them. Note that while there of course exist stable subcategories of **rcrCS** and **rCCS**, we have excluded them in order to reduce clutter. Dashed lines indicate functors which do not commute with the rest of the diagram. The definitions of the categories and functors can be seen in the following table. Those not listed in the table are intersections of other categories, e.g. CSRES is an RES which is both causal and stable, fulfilling both Definitions 7.1 and 8.2.

Proof.

1. Suppose $\mathcal{E}$ is a crRAES. Then for any reachable configuration X of $C_{ra}(\mathcal{E})$, if $X \xrightarrow{B} X \subseteq B$, then $X \setminus B \xrightarrow{B} X$ [19]. As $C_{ra}(\mathcal{E})$ is also an SCS (Proposition 7.7), we get from item 4 of Definition 3.1 of a CS and Item 2d of Definition 7.6 of an SCS, that $C_{ra}(\mathcal{E})$ is an rcrCS.
2. Suppose $\mathcal{E}$ is a CRAES, then $C_{ra}(\mathcal{E})$ is an rCCS according to [19]. □

This gives us the causal and cause-respecting categories seen in Fig. 18.

9. Conclusion

We have studied reversible event structures in a categorical setting. We have defined categories for configuration systems (CS), reversible prime event structures (RPES), reversible asymmetric event structures (RAES), and reversible general event structures (RES). We have defined functors between these categories, showing that all these event structures can be modelled as CSs and finitely enabled CSs can be modelled as RESs in a way that preserves morphisms. We also constructed coproducts for each of these categories, though products only for RESs, CSs and causal RPESs (Remark 4.22). Constructing products of RPESs and RAESs will likely be much more difficult than for RESs, since definitions of products of prime event structures are far more complex than those of general event structures [23], and products of asymmetric event structures have not been defined, to the best of our knowledge.

We showed that, unlike in the forward-only setting, reversible asymmetric event structures can be modelled as reversible general event structures, which in finitely-based cases will generate the same CS (Theorem 6.23).

We showed that our functors between RESs and finitely enabled CSs form an adjunction (Theorem 6.17), and defined a subset of finitely enabled CSs, for which our functors between RESs and finitely enabled CSs form an inverse in some cases (Theorem 6.16).

We defined stable subcategories of RESs and CSs, in which we know which events have caused each event in a given configuration. We defined a functor from stable finitely enabled CSs to stable RESs. We defined a normal form of RESs and showed that the categories of stable RESs in normal form and finitely enabled stable CSs are isomorphic (Corollary 7.15).

We also defined a normaliser function, which maps RESs to RESs in normal form which can be represented by the same CS. This significantly improves our ability to compare and combine stable systems modelled by CSs and RESs.

We have also defined cause-respecting subcategories, in which events cannot be reversed if they have caused other events, and causal subcategories, in which events can be reversed if and only if they have not caused other events.

Our longer-term aim is to formulate event structure semantics for reversible process calculi. With this in mind, we have subsequently developed a reversible variant of bundle event structures [12]. Bundle event structures were used to define semantics of LOTOS [15], and their products are simpler to construct than those of PESs. Similarly to bundle event structures, flow event structures [4] are more expressive than PESs but less so than ESs, and have a simple characterisation of their product. For this reason, a notion of reversible flow event structures would be another option for modelling reversible process calculi in the future.

Although reversible calculi are predominantly causal with few exceptions, a number include control predicates (e.g. [10]), which will need to be modelled by non-causal event structures. In [12] we have used non-causal reversible extended bundle event structures to describe semantics of CCSK with controlled reversibility. Similar work could be done with other process calculi, some of which may be expressive enough to require the use of reversible general event structures.

Acknowledgements

We thank the referees of PLACES 2017 and JLAMP for their helpful comments. This work was partially supported by EPSRC DTP award; EPSRC projects EP/K034413/1, EP/K011715/1, EP/L00058X/1, EP/N027833/1 and EP/N028201/1; EU FP7 612985 (UPSCALE); and EU COST Action IC1405 on Reversible Computation.

References

[1] C. Aubert, I. Cristescu, Contextual equivalences in configuration structures and reversibility, J. Log. Algebraic Methods Program. 86 (1) (2017) 77–106, https://doi.org/10.1016/j.jlamp.2016.08.004.
[2] S. Awodey, Category Theory, second edition, Oxf. Log. Guides, Oxford University Press, 2010.
[3] P. Baldan, A. Corradini, U. Montanari, Contextual Petri Nets, asymmetric event structures, and processes, Inf. Comput. 171 (1) (2001) 1–49, https://doi.org/10.1006/inco.2001.3060.
[4] G. Boudol, I. Castellani, Permutation of transitions: an event structure semantics for CCS and SCCS, in: J.W. de Bakker, W.P. de Roever, G. Rozenberg (Eds.), Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency, in: LNCS, vol. 354, Springer, Berlin, Heidelberg, 1989.
[5] W.J. Bowman, R.P. James, A. Sabry, Dagger traced symmetric monoidal categories and reversible programming, in: Workshop on Reversible Computation, RC 2011, 2011, https://williamjbowman.com/resources/cat-rev.pdf.
[6] I. Cristescu, J. Krivine, D. Varacca, A compositional semantics for the reversible pi-calculus, in: IEEE Symposium on Logic in Computer Science, LICS '13, IEEE Computer Society, Washington, DC, USA, 2013.
[7] I. Cristescu, J. Krivine, D. Varacca, Rigid families for the reversible π-calculus, in: Reversible Computation – Proceedings 8th International Conference, RC 2016, Bologna, Italy, July 7–8, 2016, in: Lect. Notes Comput. Sci., vol. 9720, Springer, 2016.
[8] V. Danos, J. Krivine, Reversible communicating systems, in: P. Gardner, N. Yoshida (Eds.), CONCUR, in: LNCS, vol. 3170, Springer, Berlin, Heidelberg, 2004.
[9] V. Danos, J. Krivine, Formal molecular biology done in CCS-R, Electron. Notes Theor. Comput. Sci. 180 (3) (2007) 31–49, https://doi.org/10.1016/j.entcs.2004.01.040.
[10] V. Danos, J. Krivine, P. Sobociński, General reversibility, Electron. Notes Theor. Comput. Sci. 175 (3) (2007) 75–86, https://doi.org/10.1016/j.entcs.2006.07.036.
[11] E. Graversen, I. Phillips, N. Yoshida, Towards a categorical representation of reversible event structures, in: PLACES 2017, in: EPTCS, vol. 246, 2017.
[12] E. Graversen, I. Phillips, N. Yoshida, Event structure semantics of (controlled) reversible CCS, in: J. Kari, I. Ulidowski (Eds.), RC, in: LNCS, vol. 11106, Springer International Publishing, Cham, 2018.
[13] I. Lanese, C.A. Mezzina, J.-B. Stefani, Reversing higher-order pi, in: P. Gastin, F. Laroussinie (Eds.), CONCUR, in: LNCS, vol. 6269, Springer, Berlin, Heidelberg, 2010.
[14] I. Lanese, C.A. Mezzina, F. Tiezzi, Causal-consistent reversibility, Bull. Eur. Assoc. Theor. Comput. Sci. 114 (2014) 17, https://hal.inria.fr/hal-01089350.
[15] R. Langerak, Transformations and Semantics for Lotos, Ph.D. thesis, Universiteit Twente, 1992, https://books.google.dk/books?id=qB4EAgAACAAJ.
[16] M. Nielsen, G. Plotkin, G. Winskel, Petri nets, event structures and domains, in: G. Kahn (Ed.), Semantics of Concurrent Computation, in: LNCS, vol. 70, Springer, Berlin, Heidelberg, 1979.
[17] I. Phillips, I. Ulidowski, Reversing algebraic process calculi, in: L. Aceto, A. Ingólfsdóttir (Eds.), FOSSACS, in: LNCS, vol. 3921, Springer, Berlin, Heidelberg, 2006.
[18] I. Phillips, I. Ulidowski, Reversibility and models for concurrency, Electron. Notes Theor. Comput. Sci. 192 (1) (2007) 93–108, https://doi.org/10.1016/j.entcs.2007.08.018.
[19] I. Phillips, I. Ulidowski, Reversibility and asymmetric conflict in event structures, J. Log. Algebraic Methods Program. 84 (6) (2015) 781–805, https://doi.org/10.1016/j.jlamp.2015.07.004, Special Issue on Open Problems in Concurrency Theory.
[20] I. Phillips, I. Ulidowski, S. Yuen, A reversible process calculus and the modelling of the ERK signalling pathway, in: R. Glück, T. Yokoyama (Eds.), RC, in: LNCS, vol. 7581, Springer, Berlin, Heidelberg, 2013.
[21] I. Phillips, I. Ulidowski, S. Yuen, Modelling of bonding with processes and events, in: G.W. Dueck, D.M. Miller (Eds.), RC, in: LNCS, vol. 7948, Springer, Berlin, Heidelberg, 2013.
[22] I. Ulidowski, I. Phillips, S. Yuen, Concurrency and reversibility, in: S. Yamashita, S.-i. Minato (Eds.), RC, in: LNCS, vol. 8507, Springer, Cham, 2014.
[23] F.W. Vaandrager, A Simple Definition for Parallel Composition of Prime Event Structures, CS R 8903, Centre for Mathematics and Computer Science, P.O. Box 4079, 1009 AB Amsterdam, The Netherlands, 1989, http://www.sws.cs.ru.nl/publications/papers/fvaan/CS-R8903.pdf.
[24] G. Winskel, Event structure semantics for CCS and related languages, in: M. Nielsen, E.M. Schmidt (Eds.), ICALP, in: LNCS, vol. 140, Springer, Berlin, Heidelberg, 1982.
[25] G. Winskel, Event structures, in: W. Brauer, W. Reisig, G. Rozenberg (Eds.), Petri Nets: Applications and Relationships to Other Models of Concurrency. ACPN, in: LNCS, vol. 255, Springer, Berlin, Heidelberg, 1987.