Journal of Logical and Algebraic Methods in Programming ••• (••••) •••–•••

Contents lists available at ScienceDirect

Journal of Logical and Algebraic Methods in Programming

www.elsevier.com/locate/jlamp

Reduction of event structures under history preserving bisimulation ☆

Abel Armas-Cervantes [a,*], Paolo Baldan [b,*], Luciano García-Bañuelos [a]

[a] Institute of Computer Science, University of Tartu, Estonia
[b] Department of Mathematics, University of Padua, Italy

ARTICLE INFO

Article history:
Received 17 May 2014
Received in revised form 12 October 2015
Accepted 12 October 2015
Available online xxxx

Keywords:
Prime event structures
Flow event structures
Asymmetric event structures
Reduction of event structures
History preserving bisimilarity

ABSTRACT

Event structures represent concurrent processes in terms of events and dependency relations between events like causality and conflict. Since the introduction of prime event structures, many variants of event structures have been proposed with different dependency relations and, hence, with differences in their expressive power. One of the possible benefits of using a more expressive event structure model is that of obtaining a more compact representation for the same behaviour using a smaller number of events. This article addresses the problem of reducing the size of an event structure while preserving its behaviour under a classical notion of behavioural equivalence in the true concurrency spectrum, namely history preserving bisimulation. In particular, we investigate this problem on two generalisations of prime event structures: asymmetric event structures, which rely on an asymmetric form of conflict, and flow event structures, which support a form of disjunctive causality. We single out conditions under which distinct events in an event structure can be seen as occurrences of the same activity in different contexts and thus can be folded into a single event without altering the original behaviour. By iterating the folding operation, any finite event structure can be reduced to a minimal form, behaviourally equivalent to the original one. This is not unique in general, as it depends on the order on which the folding operations are applied.

1. Introduction

The concept of concurrent process is pervasive in computer science, with applications in a multitude of distinct fields. A wide range of formalisms and techniques have been developed for the modelling and analysis of such processes. When one is interested in providing an explicit representation of the dependencies between activities, like causal dependencies, choices, possibility of parallel execution, a well established abstract model is given by event structures [1], where concurrent computations are represented by means of events and dependency relations between events. Events represent occurrences of atomic actions and dependency relations explain how events relate each other.

The seminal work of [1,2] introduces *prime event structures* (PESs), where dependencies between events are reduced to causality and conflict. An event e is a cause for e' whenever, in any computation, the occurrence of e' requires that e occurred beforehand. Events e and e' are in conflict if they never occur in the same computation.

☆ This article is a full version of the extended abstract presented at the 25th Nordic Workshop on Programming Theory, NWPT 2013, Tallinn, Estonia.

* Corresponding author.
E-mail addresses: abel.armas@ut.ee (A. Armas-Cervantes), baldan@math.unipd.it (P. Baldan), luciano.garcia@ut.ee (L. García-Bañuelos).

http://dx.doi.org/10.1016/j.jlamp.2015.10.004
2352-2208/

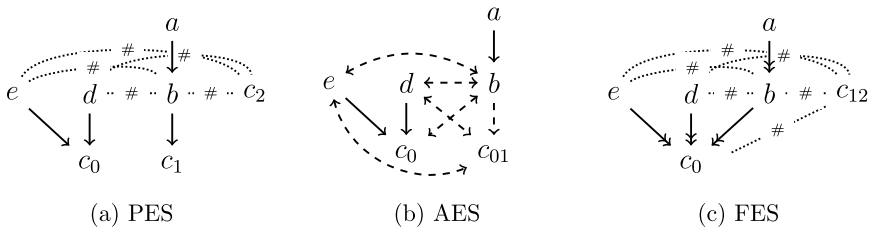


Fig. 1. Three history preserving bisimilar event structures.

Since then, many different kinds of event structures have been proposed, relying on more expressive dependency relations. In this work, we focus on two basic extensions of prime event structures, namely *asymmetric event structures* (AESs) [3], where conflict is allowed to be non-symmetric, and *flow event structures* (FESs) [4], which provide a form of disjunctive causality (the causes of an event can be chosen from a set of conflicting events).

In order to give an idea of the kind of structures the paper deals with and of the results we aim at, consider the event structures depicted in Fig. 1 in the form of graphs where nodes represent events and edges represent dependency relations. Events are labelled to represent the fact that they are instances of some abstract activities. Throughout the paper we will adopt the convention that event labels are denoted by latin letters $a, b, c, \ldots$ and, unless specified otherwise, an event is named by its label with a subscript. E.g., c_0, c_1, c_2 are distinct events, all labelled by c. Fig. 1a represents a PES. The straight directed arrows represent causality. Since causality in PESs is a transitive relation, in pictures we only depict direct causal dependencies. The annotated dotted edges represent conflict. They are undirected since conflict in PES is symmetric. For instance, the presence of a straight directed arrow from a to b indicates that a is a cause of b, written $a \leq b$, which means that *"in any computation where b occurs, event a must have occurred before"*. Instead, events d and b, connected by a dotted arrow labelled by #, are in conflict, written $b\#d$, which means that *"in any computation, either d or b does not occur"*.

Fig. 1b depicts an AES. Causal dependencies are still represented using straight directed arrows, e.g., we have that $a \leq b$. Instead, asymmetric conflict is represented by a dotted directed arrow and the corresponding relation is denoted by $\nearrow$. For instance, we have that $b \nearrow c_{01}$ which means that *"the occurrence of event c_{01} prevents b to occur afterwards"*. Hence b and c_{01} can occur in the same computation, but b has to precede c_{01} in such computations. Whenever two events, like d and b are related by asymmetric conflict in both directions, namely $d \nearrow b$ and $b \nearrow d$, then none can occur after the other, and thus they can never occur in the same computation as it happens for events in symmetric conflict in PESs.

Finally, Fig. 1c provides an example of a FES. Causality is replaced by the flow relation, which is represented with a double-headed straight arrow and denoted by $\prec$. The flow relation is not transitive. Intuitively, the flow relation expresses the set of potential direct causes for a given event. Then, in order for an event to occur, a maximal, conflict free set of potential direct causes has to occur beforehand. For instance, in the example, we have $e \prec c_0$, $d \prec c_0$ and $b \prec c_0$. Hence, $\{e, d, b\}$ is the set of potential direct causes for c_0, whose execution must be preceded by either $\{e, d\}$ or $\{b\}$.

Interestingly, it can be seen that the three event structures depicted in Figs. 1a–c represent the same set of computations, but with different numbers of events. This happens because AESs and FESs can take advantage from their greater expressiveness in order to avoid some duplication of events representing activity c. Also, it should be noted that PESs can be seen as special AESs, where asymmetric conflict is actually symmetric, and as special FESs, where the flow relation is transitive and potential causes do not contain conflicts.

The purpose of this article is to identify suitable transformations which reduce the size of AESs and FESs, without altering the original behaviour. The method is based on the identification of sets of events that intuitively represent occurrences of the same activity in different contexts and can be safely folded into a single event.

As a reference notion of behavioural equivalence we consider history preserving bisimilarity [5–7], one of the classical equivalences in true concurrency spectrum. For instance, the three event structures in Fig. 1 can be shown to be history preserving bisimilar. The AES in Fig. 1b can be obtained from the PES in Fig. 1a by folding the events c_0 and c_1 into a single event c_{01}. Similarly, the FES in Fig. 1c, can be obtained from the PES by folding events c_1 and c_2 into c_{12}.

Iterating the folding operation over a finite event structure will eventually lead to an event structure that is not further reducible, behaviourally equivalent to the original one. Unfortunately, this "minimal" event structure is not always unique and, therefore, cannot be used as a canonical representative. We will argue that the absence of a canonical minimal representative is something intrinsic, not related to limitations of our approach.

The rest of the paper is organised as follows. In Section 2 we introduce the basics of prime event structures. Furthermore, we define history preserving bisimilarity and an abstract notion of (behaviour-preserving) folding for event structures. The folding technique for AESs is presented in Section 3, while that for FESs is presented in Section 4. Finally, Section 5 draws some conclusions and proposes possible avenues for future work.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

2. Event structures, history preserving bisimilarity and foldings

In this section, we start by recalling the basics of *prime event structures*. Afterwards, we define *history preserving bisimilarity*, the reference behavioural equivalence in the paper, and introduce an abstract notion of folding, i.e., of quotient of event structures which preserves the behaviour.

First, we fix some basic notations on sets, relations and functions. Let $r \subseteq X \times X$ be a binary relation and let $Y \subseteq X$, then $r_{|Y}$ denotes the restriction of r to Y, i.e., $r_{|Y} = r \cap (Y \times Y)$. We say that r is *well-founded* if it has no infinite descending chain, i.e., if there is no sequence $\langle e_i \rangle_{i \in \mathbb{N}}$ such that $e_{i+1}\ r\ e_i$, $e_i \neq e_{i+1}$, for all $i \in \mathbb{N}$. The relation r is *acyclic* if it has no "cycles" $e_0\ r\ e_1\ r \ldots r\ e_n\ r\ e_0$ with $e_i \in X$. In particular, if r is well-founded, then it has no (non-trivial) cycles. Relation r is a *partial order* if it is reflexive, antisymmetric and transitive. Given a function $f : X \to Y$ we will denote by $f[x \mapsto y] : X \cup \{x\} \to Y \cup \{y\}$ the function defined by $f[x \mapsto y](x) = y$ and $f[x \mapsto y](z) = f(z)$ for $z \in X \setminus \{x\}$. Note that the same notation can represent an update of f, when $x \in X$, or an extension of its domain, otherwise.

2.1. Prime event structures

We recall the formal definition of *prime event structures* [1] that complements the informal description provided in the introduction. Hereafter Λ denotes a fixed set of labels.

Definition 1 *(prime event structure).* A (labelled) *prime event structure* (PES) is a tuple $\mathbb{P} = \langle E, \leq, \#, \lambda \rangle$, where E is a set of events, $\leq$ and $\#$ are binary relations on E called *causality* and *conflict*, respectively, and $\lambda : E \to \Lambda$ is a labelling function, such that

- $\leq$ is a partial order and $\lfloor e \rfloor = \{e' \in E \mid e' \leq e\}$ is finite for all $e \in E$;
- $\#$ is irreflexive, symmetric and hereditary with respect to causality, i.e., for all $e, e', e'' \in E$, if $e \# e' \leq e''$ then $e \# e''$.

Henceforth, we will write $e < e'$ for $e \leq e'$ and $e \neq e'$. In order to lighten the notation, as mentioned in the introduction, events will be often named by the corresponding labels, possibly with subscripts.

The computations of a PES are described in terms of configurations, i.e., sets of events that are closed with respect to causality and conflict free. Formally, a *configuration* of a PES $\mathbb{P} = \langle E, \leq, \#, \lambda \rangle$ is a finite set of events $C \subseteq E$ such that

- for all $e \in C$, $\lfloor e \rfloor \subseteq C$, and
- for all $e, e' \in C$, $\neg(e \# e')$.

The set of configurations of a PES $\mathbb{P}$ is denoted by $Conf(\mathbb{P})$.

2.2. History preserving bisimilarity

History preserving bisimilarity [5–7] is a classical equivalence in the true concurrency spectrum. As it happens in the interleaving approach, a bisimulation between two event structures requires any event of an event structure to be simulated by an event of the other, with the same label. Additionally, the two events are required to have the same "causal history", namely to have the same dependencies with events in their past.

In order to define history preserving bisimilarity in a way that applies uniformly to PESs, AESs and FESs, we assume to work on an abstract class of event structures, in the line of [8–10].

Definition 2 *((abstract) event structure).* An (abstract) event structure is a triple $\mathbb{E} = \langle E, Conf(\mathbb{E}), \lambda \rangle$ where E is a set of events, $Conf(\mathbb{E})$ is a set of configurations and $\lambda : E \to \Lambda$ is a labelling function. Each configuration consists of a set of events $C \subseteq E$, endowed with a partial order $\leq_C$ called the *local order* of C.

The relation $\leq_C$ associated with a configuration C intuitively represents the order in which the events in C can occur. A configuration will be often denoted simply by C, leaving the partial order $\leq_C$ implicit. An *isomorphism of configurations* $f : C_1 \to C_2$ is an isomorphism $f : C_1 \to C_2$ between the underlying sets of events that respects the order and the labelling, namely for all $e_1, e_1' \in C_1$, we have $\lambda(e) = \lambda(f(e))$ and $e_1 \leq_{C_1} e_1'$ iff $f(e_1) \leq_{C_2} f(e_1')$.

Definition 3 *(extension order).* Let $\mathbb{E}$ be an abstract event structure. The set of configurations $Conf(\mathbb{E})$ is endowed with the *extension order* defined as $C_1 \sqsubseteq C_2$ whenever $C_1 \subseteq C_2$, $\leq_{C_1} = \leq_{C_2} \cap (C_1 \times C_1)$ and for all $e_1 \in C_1$, $e_2 \in C_2$, if $e_2 \leq_{C_2} e_1$ then $e_2 \in C_1$.

Intuitively, $C_1 \sqsubseteq C_2$ means that the configuration C_1 can evolve into C_2 by executing the events in $C_2 \setminus C_1$. In fact, C_1 is required to be a subset of C_2, with events ordered exactly as in C_2 and the new events in $C_2 \setminus C_1$ cannot precede events already in C_1. This corresponds to the prefix order in [8].

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

History preserving bisimilarity can be defined on the transition systems of configurations. Given a pair of configurations $C, C' \in Conf(\mathbb{E})$ and an event $e \in E$, we write $C \xrightarrow{e} C'$ if $C' = C \cup \{e\}$ and $C \sqsubseteq C'$.

Definition 4 *(history preserving bisimilarity).* Let $\mathbb{E}_1$, $\mathbb{E}_2$ be two abstract event structures. A *history preserving (hp-)bisimulation* is a set R of triples (C_1, f, C_2), where $C_1 \in Conf(\mathbb{E}_1)$, $C_2 \in Conf(\mathbb{E}_2)$ and $f : C_1 \to C_2$ is an isomorphism of configurations, such that $(\emptyset, \emptyset, \emptyset) \in R$ and for all $(C_1, f, C_2) \in R$

a) if $C_1 \xrightarrow{e_1} C_1 \cup \{e_1\}$, for an event $e_1 \in \mathbb{E}_1$, then there exists $e_2 \in \mathbb{E}_2$ such that $C_2 \xrightarrow{e_2} C_2 \cup \{e_2\}$ and $(C_1 \cup \{e_1\}, f[e_1 \mapsto e_2], C_2 \cup \{e_2\}) \in R$;
b) if $C_2 \xrightarrow{e_2} C_2 \cup \{e_2\}$, for an event $e_2 \in \mathbb{E}_2$, then there exists $e_1 \in \mathbb{E}_1$ such that $C_1 \xrightarrow{e_1} C_1 \cup \{e_1\}$ and $(C_1 \cup \{e_1\}, f[e_1 \mapsto e_2], C_2 \cup \{e_2\}) \in R$.

When a history preserving bisimulation exists, $\mathbb{E}_1$, $\mathbb{E}_2$ are called *history preserving bisimilar*, written $\mathbb{E}_1 \sim_{hp} \mathbb{E}_2$.

Observe that the definition above ensures that an event is simulated by an event with the same label. In fact, in the triple $(C_1 \cup \{e_1\}, f[e_1 \mapsto e_2], C_2 \cup \{e_2\}) \in R$ the second component $f[e_1 \mapsto e_2]$ is an isomorphism of configurations (and thus it preserves labels).

As an example, PESs can be seen as instances of abstract event structures. Given a PES $\mathbb{P} = \langle E, \leq, \#, \lambda \rangle$ and its set of configurations $Conf(\mathbb{P})$, the local order of a configuration $C \in Conf(\mathbb{P})$ is $\leq_C = \leq_{|C}$, i.e., the restriction of the causality relation to C. The extension order turns out to be simply subset inclusion. In fact, given $C_1 \subseteq C_2$ clearly $\leq_{C_1} = \leq \cap (C_1 \times C_1)$ is the restriction to C_1 of $\leq_{C_2} = \leq \cap (C_2 \times C_2)$. Moreover, if $e_1 \in C_1$ and $e_2 \in C_2$, with $e_2 \leq_{C_1} e_1$, then necessarily $e_2 \in C_1$ since configurations are causally closed. The resulting notion of history preserving bisimilarity is the standard one in [5–7].

2.3. Foldings, abstractly

We next introduce the notion of folding, which is intended to formalise the intuition of a behaviour preserving quotient for an abstract event structure. In the next sections we will provide some concrete folding techniques for AESs and FESs.

Definition 5 *(folding).* Let $\mathbb{E}_1$ and $\mathbb{E}_2$ be event structures. A *folding morphism* is a surjective function $f : E_1 \to E_2$ such that the relation $R_f = \{(C_1, f_{|C_1}, f(C_1)) \mid C_1 \in Conf(\mathbb{E}_1)\}$ is a hp-bisimulation. A folding is called *elementary* if there is a set $X_1 \subseteq E_1$ such that for all $e_1, e_1' \in E_1$, $e_1 \neq e_1'$ and $f(e_1) = f(e_1')$ implies $e_1, e_1' \in X_1$.

In words, a folding is a mapping that "merges" sets of events of an event structure into single events, one per set, keeping the behaviour unaltered. It is elementary if it merges only a single set of events.

Sometimes, with abuse of terminology, we will refer to $\mathbb{E}_2$ as the folding of $\mathbb{E}_1$. It can be seen that under mild conditions, the target event structure is completely determined by the folding map, hence it can be seen as a sort of quotient along the map. The general theory of foldings is outside the scope of this paper, which focuses instead of the identification of some operational folding techniques.

3. Behaviour preserving reduction of AES

In this section we introduce a folding technique for asymmetric event structures. More concretely, we will describe how to identify a set of events that can be collapsed into a single event, inducing an elementary folding. This procedure can then be iterated to produce an event structure, behaviourally equivalent to the original one, that is not further reducible.

3.1. Basics of asymmetric event structures

We start by briefly reviewing the basics of asymmetric event structures, which, as mentioned before, generalise PESs by allowing a conflict relation that is not required to be symmetric.

Definition 6 *(asymmetric event structure).* A (labelled) *asymmetric event structure* (AES) is a tuple $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$, where E is a set of events, $\leq$ and $\nearrow$ are binary relations on E called *causality* and *asymmetric conflict*, respectively, and $\lambda : E \to \Lambda$ is a labelling function, such that

- $\leq$ is a partial order and $\lfloor e \rfloor = \{e' \in E \mid e' \leq e\}$ is finite for all $e \in E$;
- $\nearrow$ satisfies, for all $e, e', e'' \in E$
  1. if $e < e'$ then $e \nearrow e'$;
  2. if $e \nearrow e'$ and $e' < e''$ then $e \nearrow e''$;
  3. $\nearrow_{|\lfloor e \rfloor}$ is acyclic;
  4. if $\nearrow_{|\lfloor e \rfloor \cup \lfloor e' \rfloor}$ is cyclic then $e \nearrow e'$.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

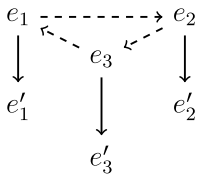


Fig. 2. Inheritance of conflict along causality in AESs.

The asymmetric conflict relation has a double interpretation, that is $e \nearrow e'$ can be understood as (i) the occurrence of e' *prevents* e to occur afterwards or (ii) the occurrence of e *precedes* the occurrence of e' in all computations where both appear. In the first view, $\nearrow$ can be seen as an asymmetric form of conflict, whence the name. Indeed, note that if e and e' are related by asymmetric conflict in both directions, i.e., $e \nearrow e'$ and $e' \nearrow e$, then none can occur after the other, and thus e and e' can never occur in the same computation as it happens for symmetric conflict in PESs. In the second view, $\nearrow$ can be seen as a weak form of causality since $e \nearrow e'$ imposes an order on the occurrences of e and e', but only when they appear in the same computation. Instead, causality $e < e'$ imposes a stricter requirement: in any computation in which e' occurs then e also occurs, and the latter must occur before.

Condition (1) of Definition 6 is motivated by the fact that, as observed in (ii) above, $\nearrow$ imposes weaker requirements than $<$, hence it is natural to ask that $\nearrow$ includes $<$. In the graphical representation of an AES, the asymmetric conflicts $e \nearrow e'$ between events that are also causally dependent $e < e'$ are not represented explicitly. Condition (2) expresses inheritance of asymmetric conflict along causality: if $e \nearrow e'$ and $e' < e''$ then e is necessarily executed before e'' when both appear in the same computation, hence $e \nearrow e''$ (see Fig. 3a). Conditions (3) and (4) can be understood by observing that events forming a cycle of asymmetric conflict cannot appear in the same computation, since each event in the cycle should occur before itself. This leads to a notion of *conflict* over sets of events $\#X$, defined by the following rules

$$\frac{e_0 \nearrow e_1 \nearrow \ldots \nearrow e_n \nearrow e_0}{\#\{e_0, \ldots, e_n\}} \qquad \frac{\#(X \cup \{e\})\ e \leq e'}{\#(X \cup \{e'\})}$$

The first rule captures the fact that events in a cycle of asymmetric conflict cannot occur in the same computation. The second rule expresses inheritance of conflict with respect to causality: if events in the set $X \cup \{e\}$ cannot occur in the same computation and $e \leq e'$, then also events in $X \cup \{e'\}$ cannot occur in the same computation. The reason is that the presence of e' requires the prior occurrence of e. Fig. 2 shows an example where $\#\{e_1, e_2, e_3\}$ by the first rule of conflict over sets and, by the second rule, applied three times, we deduce $\#\{e'_1, e'_2, e'_3\}$. Note that the second rule is essential: in fact, by Definition 6(2) we have that $e_3 \nearrow e'_1$, $e_1 \nearrow e'_2$ and $e_2 \nearrow e'_3$ (as clarified later, inherited asymmetric conflicts are often not represented in pictures), but events e'_1, e'_2, e'_3 are not in a cycle of asymmetric conflict, hence the first rule would be insufficient to prove $\#\{e'_1, e'_2, e'_3\}$.

By using the notion of conflict introduced above, condition (3) can be expressed as $\neg\#\{e\}$, for any $e \in E$. Hence it corresponds to irreflexiveness of conflict in PESs, and it ensures that any event is executable, i.e., it appears in some computation. Concerning condition (4), notice that whenever the union of the causes of e and e' includes a cycle of asymmetric conflict, according to the rules for conflict above, we have that $\#\{e, e'\}$, i.e., e and e' are in binary symmetric conflict. This will be often written $e\#e'$. In this situation, condition (4) imposes that $e \nearrow e'$ and also $e' \nearrow e$, since union is symmetric and thus the role of e and e' is interchangeable. Hence condition (4) requires that symmetric conflict is represented by asymmetric conflict in both directions.

Conditions (2) and (4) impose a form of saturation for the asymmetric conflict relation. In fact, whenever $e \nearrow e'' < e'$ or $\nearrow_{\lfloor\lfloor e \rfloor \cup \lfloor e' \rfloor}$ is cyclic, then it holds that e' cannot precede e in a computation. These conditions ask that this is also represented syntactically with an explicit asymmetric conflict. Apart from aesthetic motivations, the validity of these conditions will simplify the formulation of the folding technique.

As usual, a set of events X is called *consistent* if its causal closure does not include a subset of events in conflict, i.e., there is no $Y \subseteq \lfloor X \rfloor$ such that $\#Y$, or, equivalently, if asymmetric conflict is acyclic on $\lfloor x \rfloor$.

We recall that PESs can be seen as special AESs where asymmetric conflict is a symmetric relation. Namely, the following holds (see [3]).

Lemma 1 *(PESs are AESs). If $\mathbb{P} = \langle E, \leq, \#, \lambda \rangle$ is a PES then $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$, with $\# = \nearrow$ is an AESs. If $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ is an AESs with symmetric $\nearrow$, then $\mathbb{P} = \langle E, \leq, \#, \lambda \rangle$ with $\# = \nearrow$ is a PES.*

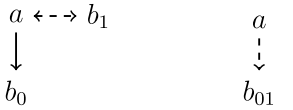
In the following, direct relations, namely causality, asymmetric conflict and conflicts that are not inherited, will play a special role.

Definition 7 *(direct relations).* Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES and let $e, e' \in E$. We say that e is an *direct cause* of e', denoted $e <_\delta e'$, when $e < e'$ and there is no e'' such that $e < e'' < e'$. An asymmetric conflict $e \nearrow e''$ is called *direct*, written $e \nearrow_\delta e'$ when there is no e'' such that $e \nearrow e'' < e'$. A binary conflict $e\#e'$ is called *direct*, written $e \#_\delta e'$, when $e \nearrow_\delta e'$ and $e' \nearrow_\delta e$.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

(a) $e \nearrow_\delta e'$ and $e \nearrow e'$ (b) $e' \nearrow_\delta e$ and $e \#_\delta e''$

Fig. 3. Inheritance of $\nearrow$.



(a) $\mathbb{A}$ (b) $\mathbb{A}$'

Fig. 4. An AES $\mathbb{A}$ and a folding $\mathbb{A}'$.

For instance, in Fig. 3a $e \nearrow_\delta e'$ while it is not the case that $e \nearrow_\delta e''$, since $e \nearrow e' < e''$. In Fig. 3b we have that $e'' \nearrow_\delta e$ and $e \nearrow_\delta e''$, hence $e \#_\delta e''$.

For the sake of readability and consistency with what we did for PESs, in pictures, often only direct relations will be represented.

Configurations in AESs are defined, as in PESs, as causally closed and conflict free sets of events.

Definition 8 *(configurations).* A *configuration* of an AES $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ is a finite set of events $C \subseteq E$ such that i) for any $e \in C$, $\lfloor e \rfloor \subseteq C$ (causal closedness) and ii) $\nearrow_{|C}$ is acyclic (conflict freeness). The set of all configurations of $\mathbb{A}$ is denoted by $Conf(\mathbb{A})$.

AESs can be seen as instances of abstract event structures by considering each configuration $C \in Conf(\mathbb{A})$ with local order $(\nearrow_{|C})^*$, i.e., the transitive closure of asymmetric conflict restricted to C. Differently from what happens for PESs, the extension order is not simply set-inclusion. It is easy to see that according to the definition in Section 2, for $C_1, C_2 \in Conf(\mathbb{A})$, we have $C_1 \sqsubseteq C_2$, iff $C_1 \subseteq C_2$ and for all $e \in C_1$, $e' \in C_2 \setminus C_1$, $\neg(e' \nearrow e)$. In words, configuration C_1 cannot be extended with events which should precede some of the events already present in C_1.

A fundamental notion is that of history of an event in a configuration.

Definition 9 *(possible histories).* Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES and let $e \in E$ be an event. Given a configuration $C \in Conf(\mathbb{A})$ such that $e \in C$, the *history* of $e \in C$ is defined as $C[\![e]\!] = \{e' \in C \mid e'(\nearrow_{|C})^* e\}$. The *set of possible histories* of e, denoted by $hist(e)$, is then defined as

$$hist(e) = \{C[\![e]\!] \mid C \in Conf(\mathbb{A}) \wedge e \in C\}$$

The history $C[\![e]\!]$ consists of the events which necessarily must occur before e in the configuration C or, in other words, it is the minimal subconfiguration of C, with respect to the extension order, which contains event e. For PESs, each event e has a uniquely determined history, which is the set $\lfloor e \rfloor$, independently of the configuration it occurs in. Instead, in the case of AESs, an event e may have several histories. For example, the event c_{02} in the AES $\mathbb{A}_2$ (Fig. 5c) has four different histories, $hist(c_{02}) = \{\{c_{02}\}, \{d, c_{02}\}, \{e, c_{02}\}, \{d, e, c_{02}\}\}$.

3.2. Quotient of AESs

The technique for behaviour preserving reduction of AESs consists in iteratively identifying a set of events carrying the same label, i.e., intuitively referring to the same activity, and replacing all the events in the set with a single event. This quotient operation is shown to induce an elementary folding, i.e., it leaves the behaviour unchanged with respect to hp-bisimilarity.

The prototypical example of folding in AESs, which exploits the expressiveness of asymmetric conflict, is provided in Fig. 4. The right AES is obtained by merging the two conflicting b-labelled events b_0 and b_1 (the conflict $b_0 \# b_1$ is inherited from $a \# b_1$). Event a is in asymmetric conflict with the event b_{01} resulting from the merge, so that $hist(b_{01})$ in $\mathbb{A}'$ includes $\{a, b_{01}\}$ and $\{b_{01}\}$, which in the AES $\mathbb{A}$ corresponds exactly to the histories of b_0 and b_1, respectively. The function mapping a identically and b_0, b_1 to b_{01} can be easily shown to be a folding.

More generally, the rough idea is that a folding will merge events in conflict, with the same label and different sets of causes, into a single event having such sets of causes as possible histories. However, events to be merged have to be chosen carefully. Consider, for instance, the AESs in Fig. 5. The AES $\mathbb{A}_1$ can be thought of as a quotient of $\mathbb{A}_0$ obtained by folding the two c-labelled events c_0 and c_1, the first in conflict with d and the second caused by d, into a single event c_{01}. The

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

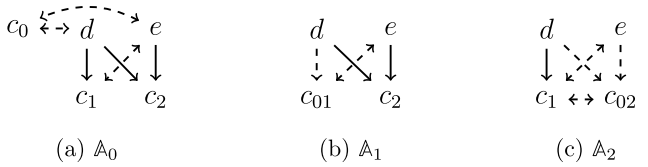


Fig. 5. AESs such that $\mathbb{A}_0 \sim_{hp} \mathbb{A}_1$ but $\mathbb{A}_0 \not\sim_{hp} \mathbb{A}_2$.

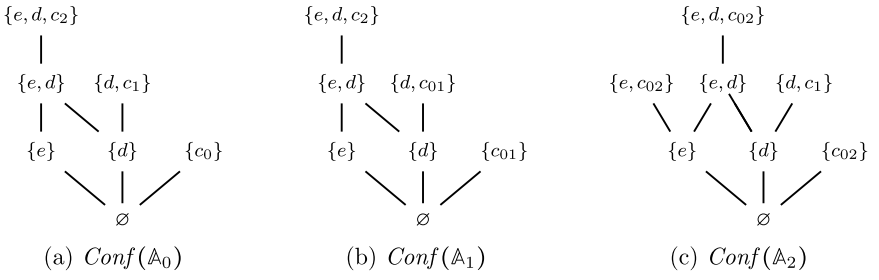


Fig. 6. The sets of configurations of the AESs in Fig. 5, ordered by extension.

dependencies $d \# c_0$ and $d < c_1$ in $\mathbb{A}_0$ give rise to the asymmetric conflict $d \nearrow c_{01}$ in $\mathbb{A}_1$. Analogously, $\mathbb{A}_2$ is obtained from $\mathbb{A}_0$ by merging c_0 and c_2 into a single event c_{02}.

Fig. 6 shows the sets of configurations of the AESs in Fig. 5, endowed with the extension order. Observe that the AESs $\mathbb{A}_0$ and $\mathbb{A}_1$ have isomorphic partially ordered sets of configurations. Instead, the poset corresponding to $\mathbb{A}_2$ has an additional configuration $\{e, c_{02}\}$ that does not correspond to any configuration in $Conf(\mathbb{A}_0)$. Hence, even though $\mathbb{A}_1$ and $\mathbb{A}_2$ are obtained from $\mathbb{A}_0$ via an apparently similar procedure, the mapping into $\mathbb{A}_1$ is a folding, while the one into $\mathbb{A}_2$ is not.

Events that can be merged, intuitively, should represent occurrences of the same activity in different contexts (leading to different causal histories for the events). Hence they surely need to have the same label and be in conflict. Additionally they should relate to the remaining events, via asymmetric conflict, essentially in the same way. This is formalised by the notion of similar events.

Definition 10 *(similar events).* Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES. We say that $X \subseteq E$ is a set of *similar events* if for all $x, x' \in X$, $e \in E \setminus X$:

1. $\lambda(x) = \lambda(x')$ and $x \# x'$;
2. if $x \nearrow e$ then $x' \nearrow e \ \vee \ e \nearrow x$;
3. $e \nearrow_\delta x \ \Rightarrow \ e \nearrow x'$.

Condition (1) requires that, as mentioned above, the events in X have the same label and are conflict. By condition (2), given two events $x, x' \in X$, if for an event $e \in E \setminus X$ we have $x \nearrow e$ then necessarily be also $x' \nearrow e$, unless $e \nearrow x$, and thus x and e are in conflict. This last clause captures the situation in which e is in the history of x' but not in that of x, and thus x and e are in conflict. Finally, condition (3) requires that any direct $\nearrow$-predecessors of an event in X remains a $\nearrow$-predecessor for all other events in X.

We next define the AES which results from the merge of a set of similar events. For a relation r on events, we will denote by $r^\forall$ and $r^\exists$ the relations between events and sets of events defined in the expected way. For instance, given an event e and a set of events X, by $e\, r^\forall\, X$ we mean that $e\, r\, x$ holds for all $x \in X$, and by $X\, r^\exists\, e$ we mean that $x\, r\, e$ holds for some $x \in X$.

Definition 11 *(quotient of an AES).* Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES and X be a set of similar events. The *quotient* of $\mathbb{A}$ with respect to X, denoted $\mathbb{A}_{/X}$, is the AES $\mathbb{A}_{/X} = \langle E_{/X}, <_{/X}, \nearrow_{/X}, \lambda_{/X} \rangle$ defined as follows

$$\begin{array}{lcl} E_{/X} & = & (E \setminus X) \cup \{e_X\} \\ \leq_{/X} & = & \leq_{|(E\setminus X)} \cup \{(e, e_X) \mid e <^\forall X\} \cup \{(e_X, e) \mid X <^\exists e\} \\ \nearrow_X & = & \nearrow_{|(E\setminus X)} \cup \{(e, e_X) \mid e \nearrow^\forall X\} \cup \{(e_X, e) \mid X \nearrow^\forall e\} \\ \lambda_{/X} & = & \lambda[e_X \mapsto \lambda(x)] \text{ for an event } x \in X. \end{array}$$

The *quotient map* $f_X : \mathbb{A} \to \mathbb{A}_{/X}$ is defined by $f_X(x) = e_X$ for $x \in X$ and $f_X(e) = e$ for $e \in E \setminus X$.

In words, the quotient of $\mathbb{A}$ is obtained by replacing the set X of events with a single event e_X, with the same label as those in X. The causes of e_X are the common causes of the events in X. Any event originally caused by at least an event

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

in X is now caused by e_X. This can be understood by recalling that the quotient map, in order to be a folding, must be in particular a simulation. Hence, on the one hand, in any computation, a common cause of all the events in X will surely occur before e_X and, on the other hand, e_X will occur before any causal consequence of an event in X. The asymmetric conflicts for e_X are exactly the common asymmetric conflicts of the events in X. This is explained by the fact that, in order to be a folding, the quotient map must preserve and reflect the local order of configurations which is given by (the transitive closure of) asymmetric conflict.

We can prove that, according to the intuition above, the quotient map is a simulation, in the sense that it preserves configurations and the extension relation on configurations. We start with a technical lemma, identifying some relevant properties of the quotient map. This will be used also to prove that $\mathbb{A}_{/X}$ is a well-defined AESs, a fact which has not been showed formally yet. It could be proved that the quotient map is an AES morphism in the sense of [3], but this has no relevant use in this paper.

Lemma 2 *(properties of the quotient map). Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES and let $X \subseteq E$ be a set of similar events. Then for all $e \in E$, $z \in E_{/X}$*

1. *if $z <_{/X} f_X(e)$ then there exists $e' \in E$ such that $e' < e$ and $f_X(e') = z$;*
2. *if $f_X(e) \nearrow_{/X} f_X(e')$ then $e \nearrow e'$;*
3. *if $e \nearrow_\delta e'$ then $f_X(e) \nearrow_{/X} f_X(e')$ or $e \# e'$;*
4. *if $e < e'$ then $f_X(e) \nearrow_{/X} f_X(e')$.*

Proof. 1. Let $z \in E_{/X}$ and $e \in E$ be such that $z <_{/X} f_X(e)$. We distinguish various cases:

- If $z = e_X$ then, by Definition 11, there exists $x \in X$ such that $x < e$. Since $f_X(x) = e_X$ and $f_X(e) = e$, this is the desired conclusion.
- If $e \in X$ (and thus $f_X(e) = e_X$) then by Definition 11, $z = e' <^\forall X$, i.e., $e' < x$ for all $x \in X$. Therefore, in particular, $e' < e$, as desired.
- If none of the above apply, then $z = e' \in E$ and $f_X(e) = e$, hence the result trivially holds.

2. Let $e, e' \in E$ and assume $f_X(e) \nearrow_{/X} f_X(e')$. If $e \in X$ and thus $f_X(e) = e_X$ then, by Definition 11, $X \nearrow^\forall e'$ and thus, again, $e \nearrow e'$. If instead, $e' \in X$ and thus $f_X(e') = e_X$ then, by Definition 11, $e \nearrow^\forall X$. Thus in particular, $e \nearrow e'$ as desired. Finally, if $e, e' \notin X$ then f_X is the identity on e, e', and thus the result trivially holds.

3. Let $e, e' \in E$ and assume $e \nearrow_\delta e'$. We distinguish three cases:

- If $e \in X$ then, by Definition 10(2), either $e' \nearrow e$ and thus $e \# e'$ and we are done, or for all $x \in X$ we have $x \nearrow e'$, namely $X \nearrow^\forall e'$. In the last case, according to Definition 11, we thus have $f_X(e) = e_X \nearrow_{/X} e' = f_X(e')$, as desired.
- If $e' \in X$ then, by Definition 10(3), for all $x \in X$ we have $e \nearrow x$, namely $e \nearrow^\forall X$. Thus, by Definition 11, $f_X(e) = e \nearrow_{/X} e_X = f_X(e')$, as desired.
- Otherwise, neither e nor e' are in X and thus the thesis trivially follows.

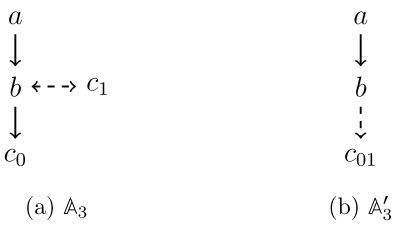
4. Let $e, e' \in E$ and assume that $e < e'$. If $e, e' \notin X$ then the relations between the two events are left unchanged. Since $e < e'$ and thus $e \nearrow e'$ we have that $f_X(e) \nearrow_{/X} f_X(e')$. If $e \in X$ then by Definition 10(2) either $x' \nearrow e'$ for all $x' \in X$ or $e' \nearrow e$. The second possibility would lead to a contradiction, since we would have $e \# e'$ and $e < e'$. Hence the first possibility must hold and thus $X \nearrow^\forall e'$, thus $f_X(e) = e_X \nearrow_{/X} f_X(e')$. Finally, if $e' \in X$, from $e < e'$ we know that $e < e'' <_\delta e'$, for some e''. By Definition 10(3), since $e'' <_\delta e'$ and thus $e'' \nearrow_\delta e'$ we have that $e'' \nearrow x$, for all $x \in X$. Recalling that $e < e''$, we have $e \nearrow x$, for all $x \in X$, namely $e \nearrow^\forall X$. Therefore $f_X(e) \nearrow_X e_X = f_X(e')$, as desired. □

Note that the converse of (2) above, i.e., if $e \nearrow e'$ then $f_X(e) \nearrow_{/X} f_X(e')$, does not hold. For instance, consider the AES in Fig. 7. If we merge c_0 and c_1, we get that $a \nearrow c_0$ but it is not true that $f_X(a) \nearrow_{/X} f_X(c_0) = c_{01}$. Moreover, note from (4) and the definition of $\leq$ in the quotient (Definition 11), it follows that the causes of some event in X, which are not common to all events, are turned into (proper) asymmetric conflicts.

Lemma 3 *($\mathbb{A}_{/X}$ is well-defined). Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES and let $X \subseteq E$ a set of similar events. Then $\mathbb{A}_{/X} = \langle E_{/X}, \leq_{/X}, \nearrow_{/X}, \lambda_{/X} \rangle$ is an AES.*

Proof. Let $\mathbb{A}_{/X} = \langle E_{/X}, \leq_{/X}, \nearrow_{/X}, \lambda_{/X} \rangle$ be defined as in Definition 11 and Let $f_X : \mathbb{A} \to \mathbb{A}_{/X}$ be the quotient map. We first note that $\leq$ is a partial order. Antisymmetry is obvious. Transitivity of $\leq_{/X}$ follows immediately by transitivity of $\leq$ in $\mathbb{A}$. Moreover, for any event $z \in E_{/X}$, we have that $\lfloor z \rfloor$ is finite. In fact, let e be any f_X-counterimage of z, i.e., $e \in E$ such that $f_X(e) = z$. For any $z' \in E_{/X}$, if $z' <_{/X} z$, by Lemma 2(1), there exists $e' < e$ such that $f_X(e') = z'$. This means that $\lfloor z \rfloor \subseteq f_X(\lfloor e \rfloor)$. Since $\lfloor e \rfloor$ is finite, we deduce that also $\lfloor z \rfloor$ is finite.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004



(a) $\mathbb{A}_3$ (b) $\mathbb{A}'_3$

Fig. 7. An AES and its quotient.

Concerning asymmetric conflict $\nearrow_{/X}$, conditions (1)–(4) in Definition 6 are easily inherited from those of $\nearrow$ in $\mathbb{A}$. More explicitly, let $z, z', z'' \in E_{/X}$. Then we have

1. If $z <_{/X} z'$ then $z \nearrow_{/X} z'$.

We distinguish various cases:

- If $z = e_X$ and $z' = f_X(e')$, for an event $e' \in E \setminus X$ then $X <^{\exists} e'$, namely, there exists $x \in X$ such that $x < e'$. This implies that $x \nearrow e'$ and thus, by the notion of similar events (Definition 10) either $x' \nearrow e'$ for all $x' \in X$ or $e' \nearrow x$. The latter possibility would lead to $x \# e'$, contradicting the fact that $x < e'$. Hence it must be $x' \nearrow e'$ for all $x' \in X$, namely $X <^{\forall} e'$, and thus $e_X \nearrow_{/X} f_X(e') = z'$.
- If $z = f_X(e)$, for an event $e \in E \setminus X$, and $z' = e_X$ then $e <^{\forall} X$. This implies that $e \nearrow^{\forall} X$ and thus $e \nearrow_{/X} e_X$.
- If both $e, e' \in E \setminus X$, the desired consequence is trivial since the relations between e and e' are not modified by the quotient operation.

2. if $z \nearrow_{/X} z' <_{/X} z''$ then $z \nearrow_{/X} z''$.

We distinguish various cases.

- If $z = e_X$ and thus $z' = f_X(e')$, $z'' = f_X(e'')$, for events $e', e'' \in E \setminus X$, then by Definition 11, we have $X \nearrow^{\forall} e'$ in $\mathbb{A}$, and thus $x \nearrow e' < e''$ for all $x \in X$. Therefore, $x \nearrow e''$ for all $x \in X$, namely $X \nearrow^{\forall} e''$ and thus $z = e_X \nearrow z'' = f_X(e'')$.
- If $z'' = e_X$ and thus $z = f_X(e)$, $z' = f_X(e')$, for events $e, e' \in E \setminus X$, then by Definition 11, we have $e <^{\forall} X$. Thus for all $x \in X$ it holds $e \nearrow e' < x$, hence $e \nearrow x$. This means that $e \nearrow^{\forall} X$ and thus $z = f_X(e) \nearrow_{/X} z'' = f_X(e_X)$, as desired.
- If $z' = e_X$ and thus $z = f_X(e)$, $z'' = f_X(e'')$, for events $e, e' \in E \setminus X$, then by Definition 11 there exists $e' \in X$ such that $e' < e''$. Moreover, $e \nearrow e'$ and thus $e \nearrow e''$ in $\mathbb{A}$. Since e, e'' are left unchanged by the quotient, $z = f_X(e) \nearrow f_X(e'') = z''$ in $\mathbb{A}_{/X}$.
- If none of $z, z', z'' \in X$ then the thesis trivially holds since the relations between such events are not modified by the quotient operation.

3. $\nearrow_{\lfloor x \rfloor_{\mathbb{A}_{/X}}}$ is acyclic for all $x \in E_{/X}$.

Let $z \in E_{/X}$ be an event and suppose that $\lfloor z \rfloor$ contains a cycle $z_1 \nearrow_{/X} z_2 \nearrow_{/X} \ldots \nearrow_{/X} z_1$. By surjectivity of f_X we can find $e \in E$ such that $z = f_X(e)$. By Lemma 2(1), there are events $e_1, \ldots, e_n \in \lfloor e \rfloor$ such that $f_X(e_i) = z_i$ for any $i \in \{1, \ldots, n\}$. By point (2) of the same lemma, $e_1 \nearrow e_2 \nearrow \ldots \nearrow e_1$. This contradicts the property of $\nearrow_{\lfloor e \rfloor} \in \mathbb{A}$ being acyclic for any event $e \in \mathbb{A}$.

4. if $\nearrow_{/X_{\lfloor \lfloor z \rfloor \cup \lfloor z' \rfloor}}$ is cyclic then $z \nearrow_{/X} z'$.

Let $e, e' \in E$ such that $f_X(e) = z$ and $f_X(e') = z'$. As observed in the proof of point (1), we have that $\lfloor z \rfloor = \lfloor f_X(e) \rfloor \subseteq f_X(\lfloor e \rfloor)$ and $\lfloor z' \rfloor = \lfloor f_X(e') \rfloor \subseteq f_X(\lfloor e' \rfloor)$. Therefore if $\nearrow_{/X}$ is cyclic over $\lfloor z \rfloor \cup \lfloor z' \rfloor$, it is cyclic also over $f_X(\lfloor e \rfloor) \cup f_X(\lfloor e' \rfloor) = f_X(\lfloor e \rfloor \cup \lfloor e' \rfloor)$. Since, by Lemma 2(2), f_X reflects asymmetric conflict, this implies that $\nearrow$ is cyclic on $\lfloor e \rfloor \cup \lfloor e' \rfloor$. Therefore $e \nearrow e'$. Since this holds for any e, e' such that $f_X(e) = z$ and $f_X(e') = z'$, a case distinction similar to that in the previous points, allows us to conclude $z \nearrow z'$. □

We can now show that the quotient map preserves configurations and the extension order.

Lemma 4 *(quotient preserves configurations). Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES, $X \subseteq E$ a set of similar events and let $f_X : \mathbb{A} \to \mathbb{A}_{/X}$ be the quotient map. Then for any configuration $C \in Conf(\mathbb{A})$ it holds that $f_X(C) \in Conf(\mathbb{A}_{/X})$ and $f_{X|C} : (C, \nearrow^*_C) \to (f_X(C), \nearrow^*_{f_X(C)})$ is an isomorphism of configurations.*

Proof. Let $C \in Conf(\mathbb{A})$ be a configuration. We first observe that $f_X(C)$ is a configuration in $Conf(\mathbb{A}_{/X})$. For proving causal closedness, take $e \in C$ and consider the event $f_X(e) \in f_X(C)$. If $z <_{/X} f_X(e)$ by Lemma 2(1) there exists $e' \in E$ such that $e' < e$ and $f_X(e') = z$. Since C is a configuration, necessarily $e' \in C$ and thus $z = f_X(e') \in f_X(C)$.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

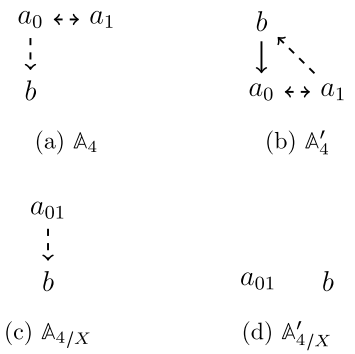


Fig. 8. Quotients with respect to the set $X = \{a_0, a_1\}$ of non-similar events.

Moreover, $\nearrow_{/X}$ is acyclic on $f_X(C)$. In fact, if there were a cycle in $f_X(C)$ it would be of the kind $f_X(e_1) \nearrow_{/X} f_X(e_2) \nearrow_{/X} \ldots \nearrow_{/X} f_X(e_n) \nearrow_{/X} f_X(e_1)$, for $e_1 \ldots, e_n \in C$. Then by Lemma 2(2), we would have $e_1 \nearrow e_2 \nearrow \ldots \nearrow e_n \nearrow e_1$, contradicting the fact that C is a configuration.

In order to prove that $f_{X|C} : (C, \nearrow_C^*) \to (f_X(C), \nearrow_{f_X(C)}^*)$ is an isomorphism of configurations, it suffices to observe that for all $e, e' \in C$ we have that

1. if $f_X(e) \nearrow_\delta f_X(e')$ then $e \nearrow e'$;
2. if $e \nearrow_\delta e'$ then $f_X(e) \nearrow f_X(e')$.

Point (1) is a special case of Lemma 2(2). For point (2), let $e \nearrow_\delta e'$. Then by Lemma 2(3), either $f_X(e) \nearrow f_X(e')$ or $e \# e'$. Since the latter cannot hold, because $e, e' \in C$ which is a configuration, necessarily $f_X(e) \nearrow f_X(e')$, as desired. □

As an immediate consequence of the above result, we can prove that the extension order is preserved and reflected by the quotient map.

Corollary 5. *Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES, $X \subseteq E$ a set of similar events and let $f_X : \mathbb{A} \to \mathbb{A}_{/X}$ be the quotient map. Then for all configuration $C, C' \in Conf(\mathbb{A})$ it holds that $C \sqsubseteq C'$ iff $f_X(C) \sqsubseteq f_X(C')$.*

Observe that conditions (2) and (3) in Definition 10 are necessary for the simulation result. For instance consider the AESs in Figs. 8a and 8b, and their quotients $\mathbb{A}_{4/X}$ and $\mathbb{A}'_{4/X}$ with respect to the set $X = \{a_0, a_1\}$, in Figs. 8c and 8d. In both cases, the quotients do not simulate the original AES.

More in detail, for the AES $\mathbb{A}_4$, we have $a_0 \nearrow b$ while neither $a_1 \nearrow b$ nor $b \nearrow a_0$, thus violating condition (2). Indeed $\mathbb{A}_4$ has the configuration $\{a_1, b\}$ with a_1 and b concurrent, which is not in the quotient. In the AES $\mathbb{A}'_4$ of Fig. 8b, $b \nearrow_\delta a_0$ while it is not the case that $b \nearrow a_1$, thus violating condition (3). In this case $\mathbb{A}'_4$ has the configuration $\{b, a_0\}$ with $b < a_0$, which is not in the quotient.

However, quotienting an AES on a set of similar events X still can alter the behaviour. Consider for instance the AESs $\mathbb{A}_0$ and $\mathbb{A}_2$ in Fig. 5. We have that $\mathbb{A}_2 = \mathbb{A}_{0/\{c_0, c_2\}}$ and $\{c_0, c_2\}$ set of similar events. We already noted that $\mathbb{A}_0$ and $\mathbb{A}_2$ are not hp-bisimilar since $\mathbb{A}_2$ admits a configuration $\{e, c_{02}\}$, which has no counterpart in $\mathbb{A}_0$: it represents a new history for a c-labelled event. The problem resides in the fact that the causes of some event $x \in X$, which are not causes for all events in X, will become asymmetric conflicts in the quotient, hence they can either appear or not in the histories of e_X. The same applies to $\nearrow$-predecessors of such causes. The (consistent) combinations of these events will lead to different possible histories for the merged event e_X. Such histories must be already histories of some event in X in the original AES, otherwise they will represent newly generated behaviours.

In order to formalise this fact, given an AES $\mathbb{A}$ and a set of similar events X, we introduce the set of possible events for X which intuitively are those events that, in the quotient, can either occur or be omitted in the histories of e_X.

Definition 12 *(possible events).* Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES and let $X \subseteq E$ a set of similar events. The set of *possible events* for X is

$$p(X) = \{e \in E \mid \neg(X \nearrow^\forall e) \wedge \neg(e <^\forall X) \ \wedge \ e \nearrow^\exists X\}.$$

According to the way in which $\nearrow_{/X}$ and $<_{/X}$ are introduced in Definition 11 the requirement $\neg(X \nearrow^\forall e)$ implies $\neg(e_X \nearrow_{/X} e)$ (and thus e_X and e are not in conflict) and the requirement $\neg(e <^\forall X)$ implies $\neg(e <_{/X} e_X)$. Finally, concerning the requirement $e \nearrow^\exists X$, namely $e \nearrow x$ for some $x \in X$, there are two possibilities. If $e \nearrow_\delta x$ then by Definition 10(3), $e \nearrow^\forall X$ and thus $e \nearrow_{/X} e_X$ in the quotient. Otherwise, $e \nearrow e' < x$ for some event e', and thus $e \nearrow_{/X} e' \nearrow_{/X} e_X$ in the quotient (since

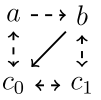


Fig. 9. The set $p(\{c_0, c_1\}) = \{a, b\}$, includes a which is neither in the history of c_0 nor of c_1.

as observed above, causalities either remain unchanged or become asymmetric conflicts). In both cases, according to the informal explanation above, they can be either included or not in the history of e_X.

Marginally, we observe that the set $p(X)$ can include events that are not in the history of any event in X. This happens for the AES in Fig. 9, taking $X = \{c_0, c_1\}$.

As mentioned above, in order not to modify the overall behaviour, all consistent subsets of $p(X)$ should match some possible history of an event in X in the original AES. For instance, in Fig. 5, in $\mathbb{A}_0$ we have that $p(\{c_0, c_1\}) = \{d\}$ while $p(\{c_1, c_2\}) = \{d, e\}$. While in the first case for any (consistent) subset of $p(\{c_0, c_1\})$ (namely $\emptyset$ and $\{d\}$) there are c-labelled events (namely c_0 and c_1) having these subsets as histories; in the second case the possible consistent subsets of $p(\{c_1, c_2\}) = \{d, e\}$ include $\{e\}$ which is not in the history of any c-labelled event. Hence the first quotient $\mathbb{A}_1 = \mathbb{A}_{0/\{c_0,c_1\}}$ preserves the behaviour, while the second $\mathbb{A}_2 = \mathbb{A}_{0/\{c_0,c_2\}}$ does not.

The above considerations lead to the notion of combinable set of events.

Definition 13 *(combinable set of events).* Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES. A set of events $X \subseteq E$ of similar events is *combinable* if for all $Y \subseteq p(X)$, consistent and causally closed (namely if $e \in Y$ and $e' \in p(X)$, $e' < e$ then $e' \in Y$) there exists $e \in X$ and $H \in hist(e)$ such that $H \cap p(X) = Y$.

We finally now show that the quotient with respect to a combinable set of events is a folding, i.e., the corresponding quotient map can be seen as a hp-bisimilarity between $\mathbb{A}$ and $\mathbb{A}_{/X}$.

Theorem 6 *(quotient map is a folding). Let $\mathbb{A} = \langle E, \leq, \nearrow, \lambda \rangle$ be an AES and let X be a combinable set of events. Then the quotient map $f_X : \mathbb{A} \to \mathbb{A}_{/X}$ is a folding.*

Proof. Let $\mathbb{A}$ be an AES, let X be a combinable set of events and let $f_X : \mathbb{A} \to \mathbb{A}_{/X}$ be the quotient map, where $\mathbb{A}_{/X} = \langle E_{/X}, \leq_{/X}, \nearrow_{/X}, \lambda_{/X} \rangle$

We prove that

$$R = \{(C_1, f_{|C_1}, f_X(C_1)) \mid C_1 \in Conf(\mathbb{A})\}$$

is a hp-bisimulation.

First of all notice that for any $C_1 \in Conf(\mathbb{A})$, if we let $C_2 = f_X(C_1)$, then by Lemma 2, $f_{|C_1} : (C_1, \nearrow^*) \to (C_2, \nearrow^*)$, is an isomorphism of pomsets.

In order to conclude, we next prove that

1. if there is $e \in E$ such that $C_1 \sqsubseteq C_1 \cup \{e\} \in Conf(\mathbb{A})$ then $C_2 \sqsubseteq C_2 \cup \{f_X(e)\} \in Conf(\mathbb{A}_{/X})$;
2. if there is $z \in E_{/X}$ such that $C_2 \sqsubseteq C_2 \cup \{z\} \in Conf(\mathbb{A}_{/X})$ then there is $e \in E$ such that $f_X(e) = z$ and $C_1 \sqsubseteq C_1 \cup \{e\} \in Conf(\mathbb{A}_{/X})$,

which corresponds to conditions (a) and (b) in Definition 4.

1. Note that $C_2 \cup \{f_X(e)\} = f_X(C_1 \cup \{e\})$ is a configuration by Lemma 4. Moreover $C_2 \sqsubseteq C_2 \cup \{f_X(e)\}$, namely there is no $e' \in C_1$ such that $f_X(e) \nearrow_{/X} f_X(e')$, otherwise by Lemma 2(2) we would have $e \nearrow e'$, contradicting $C_1 \sqsubseteq C_1 \cup \{e\}$.

2. Assume that $C_2 \sqsubseteq C_2 \cup \{z\} \in Conf(\mathbb{A}_{/X})$ for some $z \in E_{/X}$. We distinguish two cases.

2.a) $z \in E \setminus X$

Take the (unique) f_X-counterimage of e of z, namely $f_X(e) = z$. A key observation is that

$$\text{there is no } e' \in C_1 \text{ such that } e \nearrow e'. \tag{†}$$

In fact, we can show that given $e' \in C_1$ such that $e \nearrow e'$ then there exists $e'' \in C_1$ such that $z = f_X(e) \nearrow_{/X} f_X(e'')$, contradicting the fact that $C_2 \sqsubseteq C_2 \cup \{z\}$. In order to prove this, we distinguish two cases:

- First assume that $e \nearrow_\delta e'$. If $e' \notin X$ then clearly $f_X(e) \nearrow_{/X} f_X(e')$. If $e' \in X$ then by Definition 10(3) $e \nearrow x$ for all $x \in X$, namely $e \nearrow^\forall X$. Thus also in this case, by Definition 11, $f_X(e) = e \nearrow e_X = f_X(e')$. Hence the desired result holds taking $e'' = e'$.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

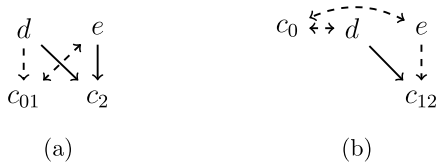


Fig. 10. Foldings for the AES in Fig. 5.

- If instead the asymmetric conflict is not direct, then there exists e''' such that $e \nearrow_\delta e''' < e'$. Since $e' \in C_1$ by causal closure also $e''' \in C$, and thus the same argument of the previous case allows to conclude.

Now we can easily prove that $C_1 \cup \{e\} \in Conf(\mathbb{A})$. For this, we need to show that $\lfloor e \rfloor \subseteq C_1$. Take any $e' < e$. Since $e \notin X$, by Definition 11, we have $f_X(e') <_{/X} f_X(e)$ and thus $f_X(e') \in f_X(C_1)$. Take $e'' \in C_1$ such that $f_X(e'') = f_X(e')$. We observe that it must necessarily be $e' = e''$. In fact, if $e' \neq e''$ it should be $e', e'' \in X$ and thus $e' \# e''$. By inheritance of conflict, this would lead to $e \# e''$ and hence $e \nearrow e''$ violating (†) above. Hence it must be $e' = e'' \in C_1$, as desired. The absence of cycles of asymmetric conflict in $C_1 \cup \{e\}$ follows immediately by the same property in C_1 and property (†) above.

Also the fact that $C_1 \sqsubseteq C_1 \cup \{e\}$ is an immediate consequence of (†) above.

2.b) $z = e_X$

Consider the set

$$Y = C_1 \cap p(X)$$

Clearly $Y \subseteq p(X)$. Moreover, it is consistent and causally closed, since Y is a subset of C_1. In fact, if $e \in Y$ and $e' \in p(X)$, $e' < e$, since $e \in Y \subseteq C_1$ and configurations are causally closed, we deduce $e' \in C_1$ and thus $e' \in Y$.

Hence, by Definition 13, there exists $x \in X$ and $H \in hist(x)$ such that $H \cap p(X) = Y$.

As in the previous case we observe that

$$\text{there is no } e \in C_1 \text{ such that } x \nearrow e. \tag{†}$$

In fact, given $e \in C_1$ such that $x \nearrow e$ then, according to Definition 10(2), we have that either $x' \nearrow e$ for all $x' \in X$ or there exists $x' \in X$ such that $\neg(x' \nearrow e)$ and $x \# e$. In the first case, we would have $X \nearrow^\forall e$ and thus $z = e_X \nearrow_{/X} f_X(e) \in C_2$, contradicting the fact that $C_2 \sqsubseteq C_2 \cup \{z\}$. In the second case, from $x \# e$ we have $e \nearrow x$ and, additionally, there is $x' \in X$ such that $\neg(x' \nearrow e)$. Hence $e \nearrow^\exists X$ and $\neg(X \nearrow^\forall e)$. Moreover it cannot be $e < x$, since $e \# x$, thus $\neg(e <^\forall X)$. This means that $e \in p(X)$. Recalling $e \in C_1$, we deduce that $e \in Y$. Since by construction $Y \subseteq H$, in turn, we get $e \in H$ which leads to a contradiction since H is a history of x, and thus it cannot include events in conflict with x.

Now observe that $\lfloor x \rfloor \subseteq C_1$. In fact for any $e < x$ either $f_X(e) < f_X(x) = e_X$ or, by Lemma 2(4), $f_X(e) \nearrow f_X(x) = e_X$. In the first case, since $f_X(e) < e_X$ necessarily $f_X(e) \in C_2$ and thus, since f_X is the identity on e, we deduce $e \in C_1$. In the second case, by Definition 11, it must be $\neg(e <^\forall X)$. Additionally, since $e < x$ he have that $e \nearrow^\exists X$ and $\neg(X \nearrow^\forall e)$ (in particular, $\neg(x \nearrow e)$). Hence $e \in p(X)$ and, since $e < x$, necessarily $e \in H$. Thus $e \in Y = H \cap p(X)$ and therefore $e \in C_1$.

By above and (†) $C_1 \cup \{x\}$ is a configuration and $C_1 \sqsubseteq C_1 \cup \{x\}$. By Lemma 4, since $f_X(C_1 \cup \{x\}) = C_2 \cup \{e_X\}$, they are isomorphic. □

By iteratively applying the quotient to a given finite AES we can thus obtain an AES which is hp-bisimilar to the original one and not further reducible. Unfortunately, this does not provide a canonical minimal representative of the behaviour. For instance, consider the AES in Fig. 5a. There exist two possible quotiented AESs, presented side-by-side in Fig. 10, which cannot be further reduced using the quotient operation.

Observe that this is not due to a limitation of our quotient technique, but rather it is intrinsic in the nature of AESs and their foldings. In fact, one can see that for these two AESs there are no non-trivial foldings (i.e., the only foldings are isomorphisms). This fact can be shown just by inspecting all the possible label preserving surjective mappings. Still, the question remains as to whether our quotient technique is in some sense complete, i.e., if it generates all the possible foldings. We will come back to this question in the conclusions.

4. Behaviour preserving reduction of FES

In this section we focus on flow event structures and, as for AESs, we propose a technique for identifying sets of events that can be collapsed into a single event, inducing an elementary folding. The basic ideas are conceptually similar to those for AESs but technically there are relevant differences.

4.1. Basics of flow event structures

We start by recalling the formal definition of (labelled) flow event structures [4].

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

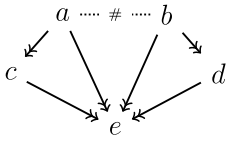


Fig. 11. A FES which is neither faithful nor full.

Definition 14 *(flow event structure).* A (labelled) *flow event structure* (FES) is a tuple $\mathbb{F} = \langle E, \#, \prec, \lambda \rangle$ where E is a set of events, $\lambda : E \to \Lambda$ is a labelling function, and

- $\prec \subseteq E \times E$, the *flow* relation, is irreflexive.
- $\# \subseteq E \times E$, the *conflict* relation, is a symmetric relation.

The $\prec$-predecessors of an event $e \in E$, are defined as $\bullet e = \{e' \mid e' \prec e\}$. Similarly, for a set of events X we write $\bullet X = \bigcup_{x \in X} \bullet x$.

Note that the flow relation is not required to be transitive. The flow predecessors $\bullet e$ of an event e can be seen as a set of possible immediate causes for e. Conflicts can exist in $\bullet e$ and, in order to be executed, e needs to be preceded by a maximal and conflict free subset of $\bullet e$. This intuition is formalised by the notion of configuration.

Definition 15 *(configuration).* Let $\mathbb{F} = \langle E, \#, \prec, \lambda \rangle$ be a FES. A *configuration* of $\mathbb{F}$ is a finite subset $C \subseteq E$ such that

1. $\neg(e \# e')$ for all $e, e' \in C$;
2. $\prec^*_{|C}$ is a partial order;
3. for all $e \in C$ and $e' \notin C$ s.t. $e' \prec e$, there exists an $e'' \in C$ such that $e' \# e'' \prec e$.

We denote by $Conf(\mathbb{F})$ the set of configurations of $\mathbb{F}$.

In words, a configuration is a conflict free subset of events, where $\prec$ is acyclic. In addition, the third condition requires that, given an event $e \in C$, for any $\prec$-predecessor $e' \prec e$ either $e' \in C$ or it is excluded by the presence of $e'' \in C$, where e'' is in conflict with e' and $e'' \prec e$. This means that for any $e \in C$, the configuration C must include a maximal consistent subset of $\prec$-predecessors of e.

FESs can be seen as instances of abstract event structures by considering each configuration $C \in Conf(\mathbb{F})$, of a FES $\mathbb{F}$, ordered by $(\prec_{|C})^*$. As for PESs, the extension order is simply subset-inclusion, namely according to the definition in Section 2, for $C_1, C_2 \in Conf(\mathbb{F})$, we have $C_1 \sqsubseteq C_2$ iff $C_1 \subseteq C_2$. In particular, observe that if $e_1 \in C_1$, $e_2 \in C_2$ and $e_2 \leq_{C_2} e_1$ then $e_2 \in C_1$. In fact, assume by contradiction that $e_2 \notin C_1$. Since $\leq_{C_2} = \prec^*_{|C_2}$, the proof can proceed on induction on the length of the $\prec$-chain connecting e_2 to e_1. If the length is 0, namely $e_2 \prec e_1$, since $e_2 \notin C_1$, by definition of configuration, there must be $e_1' \in C_1$ such that $e_1' \prec e_1$ and $e_1' \# e_2$. Since $e_1' \in C_1 \subseteq C_2$ this means that C_1 includes the conflicting events e_2, e_1', contradicting the assumption that it is a configuration. This concludes the base case. The inductive step is routine.

In FESs, the flow relation is not transitive and the conflict relation is not inherited along causal chains as in PESs. Therefore, even if two events are not in conflict syntactically, they might not appear together in any configuration. For similar reasons, an event could be not executable at all. Formally, let us define the semantic conflict relation $\#_s$ as $e \#_s e'$ when for all configurations $C \in Conf(\mathbb{F})$, it does not hold that $\{e, e'\} \subseteq C$. Then clearly $\# \subseteq \#_s$, but in general the inclusion is strict. Moreover, it could be that $e \#_s e$ for an event e (which means that e is never executable).

In line with the authors of [4], hereafter we restrict to the subclass of FESs, where:

1. semantic conflict $\#_s$ coincides with conflict $\#$ (*faithfulness*);
2. conflict is irreflexive (*fullness*), hence all events are executable;
3. $\prec$ and $\#$ are disjoint (*disjointness*).

Condition 3 is not in [4]. We assume it here since it is in line with conditions 1 and 2 and it allows us to simplify the presentation.

As an example, the FES in Fig. 11 is neither faithful nor full. For instance, despite the fact that there is no conflict $b \# c$, it holds $b \#_s c$, namely b and c cannot appear in the same configuration. In fact, since a is the only $\prec$-predecessor of c, for any configuration C, if $c \in C$ then also $a \in C$. Therefore, since $a \# b$, necessarily $b \notin C$. Similarly, $a \#_s d$ and $c \#_s d$. Additionally, observe that any configuration containing e, according to Definition 15, should include both c and d (since they are not in conflict with any other $\prec$-predecessor of e). Therefore, there is no such configuration, i.e., $e \#_s e$.

Note that FESs generalise PESs. Specifically, every PES can be seen as a special FES where the flow relation is transitive and the $\prec$-predecessors of any event are conflict free.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

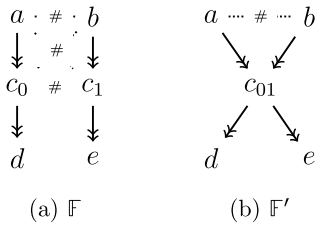


Fig. 12. A FES and a corresponding folding.

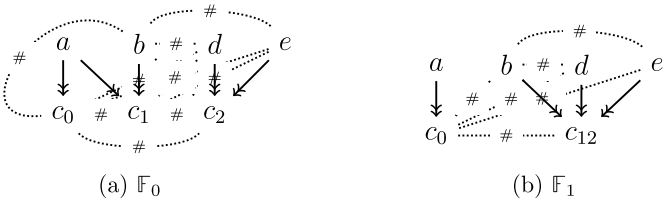


Fig. 13. Two sample FESs.

4.2. Quotient of FESs

The technique for behaviour preserving reduction of FESs, as in the case of AESs, consists in iteratively identifying a set of conflicting events with the same label that, when replaced by a single event, induces an elementary folding. As observed in the introduction, the way in which FESs generalises PESs is somehow orthogonal to that of AESs: the latter allow a non-symmetric form of conflict, while the former introduce a form of disjunctive causality. As a consequence, at a technical level the conditions defining the sets of events that can be merged are quite different.

A prototypical example of folding in FESs, which exploits the possibility of modelling disjunctive causality, is provided in Fig. 12. The FES $\mathbb{F}'$ is obtained from $\mathbb{F}$ by merging the two conflicting c-labelled events c_0 and c_1. The resulting merged event c_{01} has a and b as $\prec$-predecessors, and d and e as $\prec$-successors. Since a and b are in conflict, exactly one of them will be in a configuration including c_{01}. The function mapping a, b, d, e identically, and c_0, c_1 to c_{01} can be easily shown to be a folding.

Now consider a more complex example in Fig. 13a. First, if we take events c_0 and c_1 and try to merge them into a single event c_{01}, there would be no way of updating the dependency relations while keeping the behaviour unchanged (since b excludes c_0 and precedes c_1, the resulting dependency between b and the merged event c_{01} would be an asymmetric conflict that cannot be represented in FESs). Instead, we can merge events c_1 and c_2 in $\mathbb{F}_1$ into a single event c_{12}, thus obtaining the FES in Fig. 13b. In this case, the merge is possible because the original events c_1 and c_2 are enabled by $\{b\}$ and $\{d, e\}$, respectively, and since $b\#d$, $b\#e$, after the merge the same situation is properly represented as a disjunctive causality.

In order to define sets of events that can be safely merged we need some further notation. Given a set of events Z, we denote by $mc(Z)$ the set of maximal and consistent (i.e., conflict free) subsets of Z. Additionally, as in the case of AESs, we need to single out conflicts that are direct.

Definition 16 *(direct conflict).* Let $\mathbb{F} = \langle E, \#, \prec, \lambda\rangle$ be a FES and let $e, e' \in E$. We say that e is a *direct conflict* for e', denoted as $e \#_\delta e'$, if $e\#e'$ and $\exists Y \in mc(\bullet e)$ such that $Y \cup \{e'\}$ is consistent.

Intuitively, a conflict $e\#e'$ is direct when there is a way of reaching a configuration where e is enabled, without disabling e'. Note that direct conflict is not symmetric in FESs. For instance for $\mathbb{F}_4$ depicted in Fig. 14c, we have $e \#_\delta a_1$ while it is not the case that $a_1 \#_\delta e$.

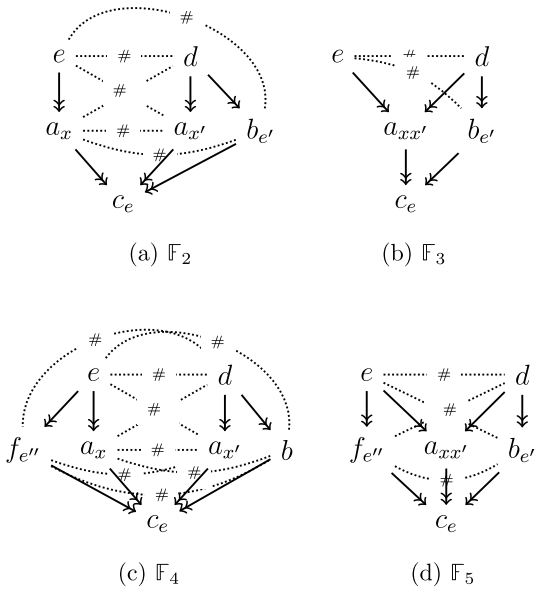
We use the extensions of relations $\#$ and $\prec$ to relations between sets and events, as already done for AESs. For instance, given $X \subseteq E$ and $e \in E$ we write $X\#^\forall e$ whenever for all $x \in X$, we have $x\#e$, or $X \prec^\exists e$ when there exists $x \in X$ such that $x \prec e$.

We can now define the notion of combinable set of events for FESs.

Definition 17 *(combinable set of events).* Let $\mathbb{F} = \langle E, \#, \prec, \lambda\rangle$ be a FES. A set of events $X \subseteq E$ is called *combinable* if for all $x, x' \in X$ and $e, e' \in E \setminus X$ the following holds

1. $\lambda(x) = \lambda(x')$ and $x\#x'$;
2. $x \#_\delta e \Rightarrow x'\#e$;
3. $x \prec e \Rightarrow x' \prec e \ \vee \ x'\#e$;
4. $e \prec x \Rightarrow \bullet x' \neq \emptyset \ \wedge \ (e \prec x' \ \vee \ (\forall e' \prec x' \wedge e' \notin \bullet x.\ e\#e'))$;

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004



(a) $\mathbb{F}_2$ (b) $\mathbb{F}_3$ (c) $\mathbb{F}_4$ (d) $\mathbb{F}_5$

Fig. 14. Example FESs to illustrate Condition 5 in Definition 17.

5. $x, e' \in {\bullet e} \ \wedge \ x \# e' \wedge \ \neg(X \# e') \Rightarrow \forall Y \in mc({\bullet e}).\ \begin{array}{l}(x \in Y \Rightarrow \exists e'' \in Y \setminus \{x\}.\ e'' \# e') \wedge \\ (X \cap Y = \emptyset \Rightarrow \exists e'' \in Y.\ X \# e'')\end{array}$.

Roughly speaking, condition (1) requires that the events in X are occurrences of the same activity (they have the same label and they are in conflict). Condition (2) requires that events in X have essentially the same conflicts: for any $x \in X$, if x is in direct conflict with an event e (hence this conflict is not derivable from the $\prec$-predecessors) then all events in X must be in conflict with e. Conditions (3) and (4) state that predecessors and successors are preserved among events in X or they can become conflicts. The rough intuition is that events whose causes are in conflict can be possibly merged thus getting a single event having the conflicting causes as $\prec$-predecessors and the conflicting consequences as $\prec$-successors. More in detail, by condition (4), if an event $x \in X$ has a non-empty set of $\prec$-predecessors, then the same must be true for all events in X. Moreover, if e is a $\prec$-predecessors of some $x \in X$ then for any other $x' \in X$, either e is a $\prec$-predecessor of x' or it is in conflict with all the $\prec$-predecessors of x' not in common with x (namely with the events in ${\bullet x'} \setminus {\bullet x}$). This ensures that, whenever we merge the events in X thus joining their $\prec$-predecessors, the maximal consistent subsets of $\prec$-predecessors will remain unchanged (see Lemma 7, where the role of condition (4) emerges formally).

Finally, condition (5) takes into account the situation in which events $x \in X$ and $e' \in E \setminus X$ are conflicting $\prec$-predecessor of an event e, but not all events in X are in conflict with e'. This is problematic because, after the merging, the conflict between x and e' will be lost, thus possibly changing the maximal subsets of $\prec$-predecessors. The condition indeed says that merging is still allowed if the conflict $x \# e'$ is not essential when forming the maximal consistent sets of $\prec$-predecessors for e. In detail, it is required that for any $Y \in mc({\bullet e})$

- if $x \in Y$ then x is not the only event in Y which is in conflict with e', so that losing the conflict $x \# e'$ would not be problematic and Y would remain a maximal consistent set;
- if none of the events of X occur in Y then this is due to the presence in Y of an event e'' in conflict with all events in X (which, in particular, is not e' and thus this will remain a maximal set even if the conflict $x \# e'$ is lost).

For example, consider the FES $\mathbb{F}_2$ in Fig. 14a. If we take $X = \{a_x, a_{x'}\}$ then condition (5) fails. Please note that events corresponding to those in condition (5) have a subscript which should suggest their role. We have ${\bullet c_e} = \{a_x, a_{x'}, b_{e'}\}$ and thus $mc({\bullet c_e}) = \{Y, Y'\}$ with $Y = \{a_x\}$ and $Y' = \{a_{x'}, b_{e'}\}$. Observe that $a_x \in Y$ but clearly there is no $e'' \in Y \setminus \{a_x\} = \emptyset$ satisfying $e'' \# b_{e'}$. The quotient of $\mathbb{F}_2$ with respect to X (formally defined later in Definition 18) would lead to the FES $\mathbb{F}_3$ in Fig. 14b, which is not behaviourally equivalent to $\mathbb{F}_2$. In particular, observe that c_e is no longer executable since it would require the prior execution of $a_{xx'}$ and $b_{e'}$, which instead cannot be in the same computation since $b_{e'} \#_s e$. This means that $a_{xx'} \#_s b_{e'}$, i.e., the two events are in semantic conflict, although it is not the case that $a_{xx'} \# b_{e'}$ (hence the quotient FES is not faithful). Note that saturating the conflict would not solve the problem. In fact, if in the quotient FES $\mathbb{F}_3$ we enforced the conflict $a_{xx'} \# b_{e'}$, then a configuration corresponding to $\{d, a_x, b_{e'}\} \in Conf(\mathbb{F}_2)$ would be missing. A situation in which condition (5) is satisfied is instead illustrated by the FES $\mathbb{F}_4$ in Fig. 14c. Again we take $X = \{a_x, a_{x'}\}$. We have ${\bullet c_e} = \{f_{e''}, a_x, a_{x'}, b_{e'}\}$ and thus $mc({\bullet c_e}) = \{Y, Y'\}$ with $Y = \{f_{e''}, a_x\}$ and $Y' = \{a_{x'}, b_{e'}\}$. Note that $a_x \in Y$ and there is indeed $f_{e''} \in Y$ such that $f_{e''} \# b_{e'}$. The

condition is satisfied also exchanging the roles of a_x and $a_{x'}$. Indeed, in the resulting quotient FES $\mathbb{F}_5$, depicted in Fig. 14d, after the execution of e or d, there are still two maximal and consistent sets of $\prec$-predecessors for the event c_e, namely $\{a_{xx'}, f_{e''}\}$ and $\{a_{xx'}, b_{e'}\}$.

We prove a technical lemma which shows that for a combinable set of events X, the maximal consistent sets of the $\prec$-predecessors of X and those of single events in X coincide. This clarifies the role of condition (4) in the definition of combinable set of events and will be useful later for proving that the quotient does not alter the behaviour.

Lemma 7 *(preservation of consistent sets). Let $\mathbb{F} = \langle E, \#, \prec, \lambda\rangle$ be a FES and let $X \subseteq E$ be a combinable set of events. Then for any consistent set $Y \subseteq E$ it holds that $Y \subseteq {\bullet}X$ iff there exists $x \in X$ such that $Y \subseteq {\bullet}x$. Hence:*

$$Y \in mc({\bullet}X) \text{ iff there exists } x \in X \text{ such that } Y \in mc({\bullet}x).$$

Proof. Let $Y \subseteq E$ be consistent. Let us assume that $Y \subseteq {\bullet}X = \bigcup_{x\in X} {\bullet}x$ and prove that there exists $x \in X$ such that $Y \subseteq {\bullet}x$. If $Y = \emptyset$ the assert is trivial. Otherwise, take $e' \in Y$. By the assumption $Y \subseteq {\bullet}X$ there must be $x' \in X$ such that $e' \in {\bullet}x'$. We show that $Y \subseteq {\bullet}x'$. In fact, for any $e \in Y$ there must exists $x \in X$ such that $e \in {\bullet}x$. Since $e \prec x$, by Definition 17(4), either $e \prec x'$ or we should have $e\#e'$. The latter possibility would contradict the consistency of Y. Hence it must be $e \prec x'$, namely $e \in {\bullet}x'$. Therefore $Y \subseteq {\bullet}x'$, as desired. The converse implication is trivial since ${\bullet}X = \bigcup_{x\in X} {\bullet}x$.

Now, the second part of the lemma, namely the fact that $Y \in mc({\bullet}X)$ iff there exists $x \in X$ such that $Y \in mc({\bullet}x)$ is an immediate consequence of the first. In fact, let $Y \in mc({\bullet}X)$. Then, by the first part of the lemma we know that there is $x \in X$ such that $Y \subseteq {\bullet}x$. Again by the first part of the lemma Y is maximal among the consistent subsets of ${\bullet}x$, since these are also consistent subsets of ${\bullet}X$. Hence $Y \in mc({\bullet}x)$. Vice versa, let $Y \in mc({\bullet}x)$. Clearly $Y \subseteq {\bullet}X$. Moreover, Y is maximal among the consistent subsets of ${\bullet}X$. To see this, take any $Y' \subseteq {\bullet}X$ consistent and assume that $Y \subseteq Y'$. By the first part of the lemma, there is $x' \in X$ such that $Y \subseteq {\bullet}x'$. Then necessarily $Y = Y'$, otherwise, by Definition 17(4), given $y' \in Y' \setminus Y$ we would have $y'\#y$ for any $y \in Y$, which is absurd since $Y \subseteq Y'$ and Y' consistent. □

We next formally define the quotient of a FESs with respect to a combinable set of events.

Definition 18 *(quotient of FESs).* Let $\mathbb{F} = \langle E, \#, \prec, \lambda\rangle$ be a FES, X be a combinable set of events. The *quotient* of $\mathbb{F}$ with respect to X, denoted by $\mathbb{F}_{/X}$, is the FES $\mathbb{F}_{/X} = \langle E_{/X}, \#_{/X}, \prec_{/X}, \lambda_{/X}\rangle$ where

$$\begin{array}{lcl} E_{/X} & = & (E \setminus X) \cup \{e_X\} \\ \#_{/X} & = & \#_{|(E\setminus X)} \cup \{(e, e_X) \mid e\#^{\forall} X\} \\ \prec_{/X} & = & \prec_{|(E\setminus X)} \cup \{(e, e_X) \mid e \prec^{\exists} X\} \cup \{(e_X, e') \mid X \prec^{\exists} e'\} \\ \lambda_{/X} & = & \lambda_{/X}[e_X \mapsto \lambda(x)] \text{ for an event } x \in X. \end{array}$$

The *quotient map* $f_X : \mathbb{F} \to \mathbb{F}_{/X}$ is defined by $f_X(x) = e_X$ for $x \in X$ and $f_X(e) = e$ for $e \in E \setminus X$.

The rest of the section is dedicated to showing that the quotient operation on FESs induces a (elementary) folding, namely it preserves hp-bisimilarity.

The idea underlying the proof for AESs was that merged events are occurrences of the same activity with different histories. They could be merged if their histories were compatible and, after merging, the possible histories remained the same. For FESs the intuition of the proof is similar, but now events can occur after a maximal consistent set of $\prec$-predecessors which roughly play the role of histories in AESs. By Lemma 7, after merging a set of combinable events this maximal subsets of consistent events remains unchanged. This will be a core ingredient in the proof that the quotient does not alter the behaviour.

We start by showing some properties of the quotient map which will be used later for showing that it transforms configurations of the original FES into isomorphic configurations of the quotient FES. We do not rely on the notion of FES morphism from [11], which would be too strong for our needs (in particular, condition (iii) of [11, Definition 4] is not satisfied by our quotient map).

Lemma 8 *(properties of the quotient map). Let $\mathbb{F} = \langle E, \#, \prec, \lambda\rangle$ be a FES, $X \subseteq E$ be a combinable set of events let $f_X : \mathbb{F} \to \mathbb{F}_{/X}$ be the quotient map. Then for all $e, e' \in E$:*

1. *if $f_X(e)\#_{/X} f_X(e')$ then $e\#e'$;*
2. *if $e \prec e'$ then $f_X(e) \prec_{/X} f_X(e')$;*
3. *if $f_X(e) \prec_{/X} f_X(e')$ then $e \prec e' \vee e\#e'$;*
4. *if $f_X(e) = f_X(e')$ then $e = e' \vee e\#e'$.*

Proof. 1. Let $e, e' \in E$ and assume $f_X(e)\#_{/X} f_X(e')$. Notice that at least one between e and e' is not in X, otherwise we would have $f_X(e) = f_X(e')$ that is a contradiction since, by construction, $\#_{/X}$ is irreflexive. We distinguish various cases. If $e \in X$ and thus $f_X(e) = e_X$, then by definition of conflict in the quotient FES (Definition 18), since $f_X(e) = e_X\#_{/X} f_X(e')$, it

must be $X\#^{\forall}e'$, and thus in particular $e\#e'$, as desired. The case in which $e' \in X$ is analogous, since conflict is symmetric. Otherwise, if $e, e' \notin X$ the property trivially holds, since f_X is the identity on e, e' and their mutual relations are not changed by the quotient operation.

2. Let $e, e' \in E$ be such that $e \prec e'$. Note that it cannot be $e, e' \in X$, otherwise, we would have $e \prec e'$ and, by Definition 17(1), $e\#e'$, violating the disjointness of $\prec$ and $\#$. Hence we distinguish the following cases:

- If $e \in X$ and $e' \notin X$, by Definition 18, $e_X = f_X(e) \prec_{/X} f_X(e') = e'$ as desired.
- If $e' \in X$ and $e \notin X$, by construction, $e = f_X(e) \prec_{/X} f_X(e') = e_X$.
- If $e, e' \notin X$ then f_X is the identity on e, e' and the result trivially holds.

3. Let $e, e' \in E$ be such that $f_X(e) \prec_{/X} f_X(e')$. Note that it cannot be $e, e' \in X$, otherwise, we would have $f_X(e) = e_X \prec_{/X} e_X = f_X(e')$, while by construction $\prec_{/X}$ is irreflexive. Hence we distinguish the following cases:

- If $e \in X$ and $e' \notin X$, by construction, there exists $x' \in X$ such that $x' \prec e'$. Then, either $x' = e$ and thus $e \prec e'$, or, by Definition 17(3), $e'\#e$ as desired.
- If $e' \in X$ and $e \notin X$, by construction, there exists $x \in X$ such that $e \prec x$. Then, either $x = e'$ and thus $e \prec e'$, or, by Definition 17(4), $e'\#e$ as desired.
- Otherwise, if $e, e' \notin X$ then f_X is the identity on e, e' and hence $e \prec e'$.

4. Let $e, e' \in E$ such that $f_X(e) = f_X(e')$, with $e \neq e'$. This means that $e, e' \in X$ and thus, since the events in X are pairwise conflicting, we have that $e\#e'$. □

We can now show that the quotient map transforms any configuration of the original FES into an isomorphic configuration of the quotient.

Lemma 9 *(quotient preserves configurations). Let $\mathbb{F} = \langle E, \#, \prec, \lambda\rangle$ be a FES, $X \subseteq E$ be a combinable set of events and let $f_X : \mathbb{F} \to \mathbb{F}_{/X}$ be the quotient map. For any configuration $C \in Conf(\mathbb{F})$ then $f_X(C) \in Conf(\mathbb{F}_{/X})$ and, additionally, $f_{X|C} : (C, \prec_C^*) \to (f_X(C), \prec_{f_X(C)}^*)$ is an isomorphism of configurations.*

Proof. We first prove that $f_X(C)$ is a configuration.

1. $f_X(C)$ is conflict free.
   This follows directly from Lemma 8(1). In fact, for $e, e' \in C$, if it were $f_X(e)\#_{/X} f_X(e')$ then we would deduce $e\#e'$, contradicting the fact that C is a configuration.
2. $f_X(C)$ has no $\prec$-cycles.
   Observe that, by Lemma 8(3), f_X reflects the flow relation over events of a configuration, namely for $e, e' \in C$, if $f_X(e) \prec_{/X} f_X(e')$ then $e \prec e'$ (since the case $e\#e'$ would contradict the fact that C is a configuration). As a consequence, a $\prec$-cycle in $f_X(C)$ would be reflected in C.
3. For all $z \in f_X(C)$ and $z' \notin f_X(C)$ s.t. $z' \prec z$, there exists $z'' \in f_X(C)$ such that $z'\#z'' \prec z$.
   Let $z \in f_X(C)$, $z' \notin f_X(C)$, such that $z' \prec z$. Therefore, there are $e \in C$ such that $z = f_X(e)$ and, by surjectivity of f_X, $e' \notin C$ such that $z' = f_X(e')$.
   By Lemma 8(3) either (i) $e'\#e$ or (ii) $e' \prec e$. Below we treat the two cases separately.
   (i) If $e'\#e$, the fact that $\neg(e' \prec e)$ while $f_X(e') \prec_{/X} f_X(e)$, the construction in Definition 18, implies that one of the following holds:
   - $e \in X$ and there exists $x \in X$ such that $e' \prec x$.
     Note that the conflict $e\#e'$ cannot be direct, otherwise, by Definition 17(2), one should have also $x\#e'$. Hence, since by definition of configuration, the set $\bullet e \cap C \in mc(\bullet e)$, there must be $e'' \in \bullet e \cap C$ such that $e'\#e''$. Hence $e'' \in C$ and $e'' \prec e$. Therefore by Lemma 8(2), $f_X(e'') \prec_{/X} f_X(e) = z$. Moreover, since $e', e'' \notin X$, we have $f_X(e'')\#_{/X} f_X(e') = z'$, as desired.
   - $e' \in X$ and there exists $x' \in X$ such that $x' \prec e$.
     In this case note that $f_X(x') = f_X(e') = e_X$ and thus we can take x' instead of e', and proceed as in case (ii).

   (ii) Let us focus on the other case, in which $e' \prec e$. Since C is a configuration, there exists $e'' \in C$ such that $e'' \prec e$ and $e''\#e'$. By Lemma 8(2), $f_X(e'') \prec f_X(e) = z$. We distinguish various subcases:
   (a) $\{e', e''\} \subseteq X$. This simply cannot happen as it would imply $f_X(e') = f_X(e'') \in f_X(C)$, while we are assuming $f_X(e') \notin f_X(C)$.
   (b) $e' \in X, e'' \notin X$. Let $Y \in mc(\bullet e)$ be the set of maximal and consistent set of predecessors of e in C. Obviously, $e'' \in Y$ and, by Lemma 8(2), for all $e_1 \in Y$ we have $f_X(e_1) \prec f_X(e) = z$ and $f_X(e_1) \in f_X(C)$. Clearly, there is no $e_2 \in Y \cap X$ such that $e_2 \in C$, otherwise $f_X(e_2) = f_X(e') = z' \in f_X(C)$ and this would contradict the assumptions.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

Therefore, $Y \cap X = \emptyset$ and, by Definition 17(5), there exists $e_1'' \in Y$ such that $e_1''\#^{\forall}X$. In this case, by construction, $f_X(e_1'')\#_{/X} f_X(e') = z' = e_X$ and, since $f_X(e_1'') \in f_X(C)$, this gives the desired result.

(c) $e' \notin X, e'' \in X$. By Definition 17(5), for all $Y \in mc(\bullet e)$, with $e'' \in Y$ there is $e_1 \in Y \setminus \{e''\}$ such that $e_1\#e'$. Since neither e' nor e_1 are in X, this conflict is preserved by the quotient map and thus $f_X(e_1)\#f_X(e') = z'$. Since, $f_X(e_1) \in f_X(C)$ and, by Lemma 8(2), $f_X(e_1) \prec f_X(e) = z$, we get the desired results.

(d) $\{e', e''\} \nsubseteq X$. Since $\{e', e''\} \nsubseteq X$ and $e'\#e''$ then, by Lemma 8(1), $f_X(e'')\#f_X(e')$, as desired.

Concerning the last assertion, note that the fact that $f_{X\lfloor C} : (C, \prec_C^*) \to (f_X(C), \prec^*_{f_X(C)})$ is an isomorphism follows immediately by items (2) and (3) of Lemma 8. □

Recall that FESs are assumed to be faithful, full and disjoint. We next prove that the quotient preserves these properties.

Lemma 10 *(quotient is full and faithful). Let $\mathbb{F} = \langle E, \#, \prec, \lambda\rangle$ be a FES, X a combinable set of events and let $f_X : \mathbb{F} \to \mathbb{F}_{/X}$ be the quotient map. The FES $\mathbb{F}_{/X}$ is 1) faithful, 2) full and 3) disjoint.*

Proof. 1. Let $z, z' \in E_{/X}$ be events in $\mathbb{F}_{/X}$ such that $\neg(z\#z')$. We need to prove that there exists a configuration $C_1 \in Conf(\mathbb{F}_{/X})$ such that $\{z, z'\} \subseteq C_1$.

Take $e, e' \in E$ such that $f_X(e) = z$ and $f_X(e') = z'$ (they exist since f_X is surjective). If $\neg(e\#e')$ then, by faithfulness of $\mathbb{F}$, there exists $C_0 \in Conf(\mathbb{F})$ such that $\{e, e'\} \subseteq C_0$. By Lemma 9, $f_X(C_0) \in Conf(\mathbb{F}_{/X})$ is the desired configuration, since $\{z, z'\} = \{f_X(e), f_X(e')\} \subseteq f_X(C_0)$.

If instead $e\#e'$, it means that one of the two events is in X (otherwise their dependencies would not be changed by the quotient). Assume without loss of generality that $e \in X$, hence $z = e_X$, and $e' \notin X$. The fact that $\neg(f_X(e)\#f_X(e'))$ means that there is $e'' \in X$ such that $\neg(e''\#e')$. Therefore, again by fullness there exists $C_0 \in Conf(\mathbb{F})$ such that $\{e'', e'\} \subseteq C_0$ and we conclude as above. In fact, $f_X(e'') = f_X(e) = z$, hence $\{z, z'\} = \{f_X(e), f_X(e')\} \subseteq f_X(C_0)$, which is a configuration by Lemma 9.

2. By Lemma 8(1) and surjectivity of f_X, a self-conflicting (inconsistent) event in $\mathbb{F}_{/X}$ would be reflected in $\mathbb{F}$. More precisely, let $z \in \mathbb{F}_{/X}$ such that $z\#z$. Then take $e \in \mathbb{F}$ such that $f_X(e) = z$. We have $f_X(e)\#f_X(e)$ and thus, by Lemma 8(1), $e\#e$, contradicting the fullness of $\mathbb{F}$.

3. In order to show that $\#_{/X}$ and $\prec_{/X}$ are disjoint we proceed by contradiction. Assume that $z \prec_{/X} z'$ and $z\#_{/X}z'$. By Definition 18 there are $e, e' \in E$ such that $f_X(e) = z$, $f_X(e') = z'$ and $e \prec e'$. However, by Lemma 8(1), we have also $e\#e'$, contradicting the disjointness of $\mathbb{F}$. □

Building on the previous technical results, we can finally prove that the quotient map f_X is a folding, i.e., that it can be seen as a hp-bisimulation.

Theorem 11 *(quotient map is a folding). Let $\mathbb{F} = \langle E, \#, \prec, \lambda\rangle$ be a FES and let $X \subseteq E$ be a combinable set of events. Then the quotient map $f_X : \mathbb{F} \to \mathbb{F}_{/X}$ is a folding.*

Proof. Let $\mathbb{F}$ be a FES, X be a combinable set of events and $f_X : \mathbb{F} \to \mathbb{F}_{/X}$ be the quotient map, where $\mathbb{F}_{/X} = \langle E_{/X}, \#_{/X}, \prec_{/X}, \lambda_{/X}\rangle$. We prove that

$$R = \{(C_1, f_{X\lfloor C_1}, f_X(C_1)) \mid C_1 \in Conf(\mathbb{F})\}$$

is a hp-bisimulation.

Given a configuration $C_1 \in Conf(\mathbb{F})$, define $C_2 = f_X(C_1)$. Recall from Lemma 9 that $f_{X\lfloor C_1} : (C_1, \prec^*) \to (C_2, \prec^*)$ is an isomorphism of pomsets.

In order to show that R is a hp-bisimilarity it remains to prove that

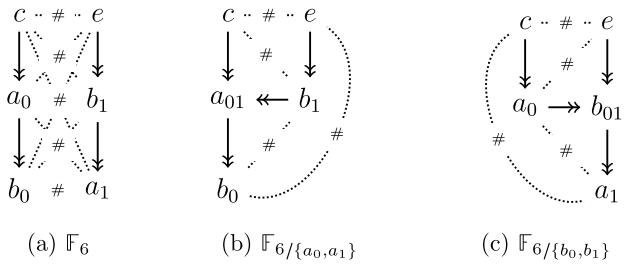
1. if there is $e \in E$ such that $C_1 \cup \{e\} \in Conf(\mathbb{F})$ then $C_2 \cup \{f_X(e)\} \in Conf(\mathbb{F}_{/X})$;
2. if there is $z \in E_{/X}$ such that $C_2 \cup \{z\} \in Conf(\mathbb{F})$ then there is $e \in E$ such that $f_X(e) = z$ and $C_1 \cup \{e\} \in Conf(\mathbb{F})$.

In fact, since the extension order for FESs is subset inclusion, (1) and (2) above correspond to conditions (a) and (b) in Definition 4.

We prove the two points separately:

1. The fact that if $C_1 \cup \{e\} \in Conf(\mathbb{F})$ then $C_2 \cup \{f_X(e)\} \in Conf(\mathbb{F}_{/X})$ follows immediately by Lemma 9, since $C_2 \cup \{f_X(e)\} = f_X(C_1 \cup \{e\})$.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004



(a) $\mathbb{F}_6$ (b) $\mathbb{F}_{6/\{a_0,a_1\}}$ (c) $\mathbb{F}_{6/\{b_0,b_1\}}$

Fig. 15. A FES and two minimal non-isomorphic quotients.

2. Let $z \in E_{/X}$ be such that $C_2 \cup \{z\} \in \mathit{Conf}(\mathbb{F}_{/X})$ and let us show that there is an event $e \in E$ such that $f_X(e) = z$, $C_1 \cup \{e\} \in \mathit{Conf}(\mathbb{F})$.
Let $Y_2 = {\bullet}z \cap C_2$ be the set of $\prec$-predecessors of z in C_2. By definition of configuration in FESs we know that $Y_2 \in \mathit{mc}({\bullet}z)$.
We distinguish two cases:
   (a) $z = e_X$.
   In this case events in ${\bullet}z$ are left unchanged by the quotient and hence if we let $Y_1 = Y_2$ we have that $Y_1 \subseteq C_1$, $f_X(Y_1) = Y_2$ and Y_1 is consistent. By definition of the quotient (Definition 18) we have that $e \prec_{/X} e_X$ iff $e \prec^{\exists} X$ and hence $Y_1 \subseteq {\bullet}X$ and, by Lemma 7, there is an event $e' \in X$, s.t. $Y_1 \in \mathit{mc}({\bullet}e')$. Since $Y_1 \subseteq C_1$, we deduce that $C_1 \cup \{e'\} \in \mathit{Conf}(\mathbb{F})$, with $f_X(e') = e_X$, as desired.

   (b) $z \neq e_X$.
   In this case the event $z = e \in E \setminus X$ is mapped identically by the quotient map f_X. In order to conclude, we just need to show that $C_1 \cup \{e\}$ is a configuration. Let $Y_1 = \{e' \in C_1 \mid f_X(e') \in Y_2\}$.
   We have that $Y_1 \subseteq {\bullet}e$. In order to prove this fact, note that for any $e' \in Y_1$, since $f_X(e') \prec_{/X} f_X(e) = z$, by Lemma 8(3) we know that $e' \prec e$ or $e' \# e$. We show that the second case cannot happen. If $e' \notin X$ this is obvious. Otherwise, if $e' \in X$, from $\neg(f_X(e) \#_{/X} f_X(e'))$, by Definition 18, there is $x \in X$ such that $\neg e \# x$. Then by Definition 17(2), the conflict $e' \# e$ is not direct. Therefore, since ${\bullet}e' \cap C_1 \in \mathit{mc}({\bullet}e')$, by definition of direct conflict, there is $e'' \in {\bullet}e' \cap C_1$ such that $e'' \# e$. Since $e'' \notin X$, this conflict is preserved by the quotient map and we get that $f_X(e'') \#_{/X} f_X(e)$, which is absurd since $f_X(e), f_X(e'') \in f_X(C_1) \cup \{z\}$, and the latter is a configuration by hypothesis.
   The set Y_1 is clearly consistent, since it is included in C_1. It is also maximal, i.e., $Y_1 \in \mathit{mc}({\bullet}e)$. In fact if it were not maximal, there would be $e'' \in {\bullet}e \setminus Y_1$ such that $Y_1 \cup \{e''\}$ is consistent. But then, since the quotient map preserves configurations and thus consistent sets, $f_X(Y_1 \cup \{e''\})$ would be consistent and strictly larger than Y_2.
   Since $Y_1 \in \mathit{mc}({\bullet}e)$, we conclude that $Y_1 \cup \{e\}$ is a configuration, as desired. □

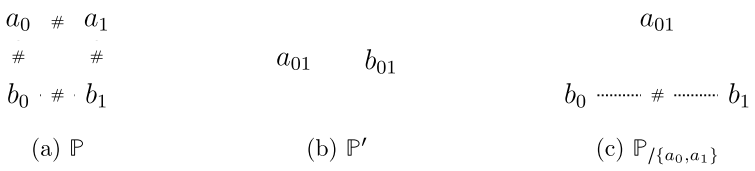
As in the case of AESs the iterative application of the quotient operation to a finite FES leads to a "minimal" FES hp-bisimilar to the original one. Different sequences of quotient operations can lead to non-isomorphic FESs, which are not further reducible. An example is provided in Fig. 15. In the FES $\mathbb{F}_6$ there are two combinable sets of events, namely $\{a_0, a_1\}$ and $\{b_0, b_1\}$. In the quotient $\mathbb{F}_{6/\{a_0,a_1\}}$, depicted in Fig. 15b, the set $\{b_0, b_1\}$ is no longer combinable. In fact, condition (4) in Definition 17 is violated since $a \prec b_0$, but it does not hold that $a \prec b_1$ and there is event e such that $e \prec b_1$, $\neg(e \prec a)$ and $\neg(a \# e)$. Similarly, in the quotient $\mathbb{F}_{6/\{b_0,b_1\}}$, depicted in Fig. 15c, the set $\{a_0, a_1\}$ is no longer combinable. Hence we get two non-isomorphic FESs which are not further reducible. Also in this case, one can see that this is intrinsic in the nature of FESs and their foldings. In fact, by inspecting all the possible label preserving surjective mappings one realises that the two quotients do not admit any non-trivial folding.

5. Conclusion and future work

This paper presents reduction techniques for AESs and FESs, which are aimed at reducing the size of an event structure without altering its behaviour. The techniques are based on suitably defined quotient operations. Each quotient merges a set of events that represent instances of the same activity in different contexts. The equivalence notion adopted is history preserving bisimulation, a standard equivalence in the true concurrency spectrum. Due to the different expressive power of AESs and FESs, tailored quotient techniques have been proposed for the two brands of event structures.

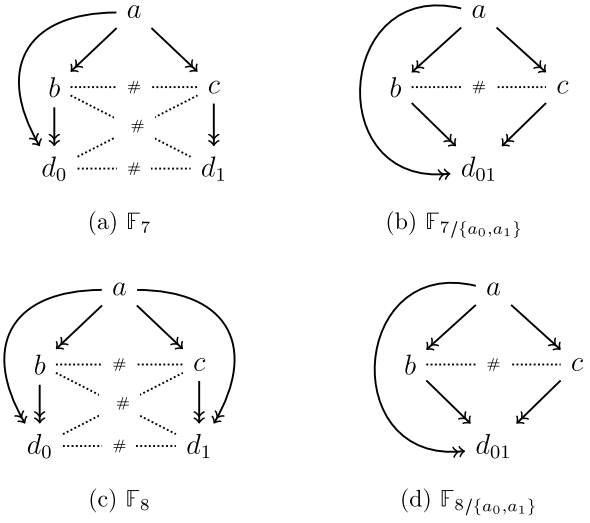
In the paper, we first provide an abstract notion of behaviour preserving quotient, referred to as folding. Then we study suitable conditions which identify sets of events that can be safely merged, in a way that induces (elementary) foldings. A natural question arises in this context, concerning the "completeness" of the quotient techniques we identified. More precisely, is any folding induced by a sequence of quotient operations? The answer is negative. In fact, consider the PES in Fig. 16a, which can either be seen as an AES or a FES. It admits the folding in Fig. 16b, where a_0, a_1 are merged into a_{01} and similarly b_0, b_1 are merged into b_{01}. It is not difficult to see that $\mathbb{P}'$ cannot be obtained by our quotient operations, neither seeing $\mathbb{P}$ as an AES nor as a FES.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004



(a) $\mathbb{P}$ (b) $\mathbb{P}'$ (c) $\mathbb{P}_{/\{a_0,a_1\}}$

Fig. 16. A PES $\mathbb{P}$ and a possible folding $\mathbb{P}'$ that cannot be obtained by composing elementary foldings.



(a) $\mathbb{F}_7$ (b) $\mathbb{F}_{7/\{a_0,a_1\}}$ (c) $\mathbb{F}_8$ (d) $\mathbb{F}_{8/\{a_0,a_1\}}$

Fig. 17. Non-completeness of the quotient technique for FESs.

The limitation seems to reside in the fact that a quotient operation realises only elementary foldings (only a single set of events is merged each time). Indeed, the folding $\mathbb{P}'$ cannot be expressed as the composition of elementary foldings. For instance, notice that the quotient $\mathbb{P}_{/\{a_0,a_1\}}$ in Fig. 16c is not a folding. In fact event a_0 should be simulated by $f_{\{a_0,a_1\}}(a_0) = a_{01}$. However, in $\mathbb{P}_{/\{a_0,a_1\}}$ after a_{01} we can execute b_0; whereas in $\mathbb{P}$ once a_0 occurs then event b_0 is ruled out.

Some preliminary results lead us to conjecture that indeed the quotient technique for AESs is complete for elementary foldings and a complete technique for general foldings can be defined at the price of reducing the efficiency (all sets to be merged have to be searched for at the same time). For FESs, instead, the intensional nature of the dependency relations seems to be an obstacle toward a completeness result already for elementary foldings.

In Fig. 17 two FESs $\mathbb{F}_7$ and $\mathbb{F}_8$ are depicted along with corresponding quotients with respect to the set $X = \{d_0, d_1\}$. It is not difficult to see that the two FESs have exactly the same posets of configurations. Indeed, the only difference between $\mathbb{F}_7$ and $\mathbb{F}_8$ is the absence, in the former, of the flow $a \prec d_1$. Since a is the only $\prec$-predecessor of c, which in turn is the only $\prec$-predecessor of d_1, this flow is semantically enforced. Hence, its explicit presence does not alter, in any way, the behaviour. However, this subtle syntactic difference is very important for the quotient operation. It is immediate to see that $\{d_0, d_1\}$ is combinable in $\mathbb{F}_8$; whereas, $\{d_0, d_1\}$ is not combinable in $\mathbb{F}_7$, because condition (4) of Definition 17 is violated. In fact, in $\mathbb{F}_7$, we have $a \prec d_0$, but $\neg(a \prec d_1)$ and there is $c \prec d_1$ such that $\neg(c \prec d_0)$ and $\neg(c\#a)$. Still, the quotient operation applied to $\mathbb{F}_7$ and $\mathbb{F}_8$ produces the same result $\mathbb{F}_{7/\{d_0,d_1\}} = \mathbb{F}_{8/\{d_0,d_1\}}$. Thus, in both cases, the quotient preserves the behaviour, namely it induces an elementary folding, but only the second is allowed by our technique. This means that, in the case of FESs, completeness fails also for elementary foldings. We conjecture that this problem can be faced by restricting to classes of FESs where the dependency relations are saturated (in the spirit of the faithfulness and fullness requirements).

An interesting line of future research is the development of a general theory of foldings, addressing elementary and non-elementary folding techniques, answering in a systematic way to these completeness questions.

It turned out that neither AESs nor FESs offer a canonical representation of the behaviour of a process. More specifically, the same process can have non-isomorphic minimal foldings both in the case of AESs and FESs. Therefore, a natural venue for future work is to investigate how to characterise an ordering on foldings, leading to a notion of minimal canonical AESs or FESs. This issue is investigated in [12].

We noted that the conditions defining sets of combinable events are essentially orthogonal for AESs and FESs. In this respect, we envision a transformation from AESs to FESs which would allow further folding at the price of inserting unobservable events to simulate asymmetric conflict on a FES. Such a transformation could open the possibility of taking advantage of the combined expressiveness of AES and FES, possibly leading to more compact representations. This is therefore another venue for future research.

Please cite this article in press as: A. Armas-Cervantes et al., Reduction of event structures under history preserving bisimulation, Journal of Logical and Algebraic Methods in Programming (2015), http://dx.doi.org/10.1016/j.jlamp.2015.10.004

Concerning the related literature, the minimisation of the behaviour of a process can be translated into some kind of minimisation problem for automata or labelled transition system. Most available techniques focus on interleaving behavioural equivalences (like language or trace equivalence or various forms of bisimilarity). We are not aware of approaches for the minimisation of event structures or partially ordered models of computation. In some cases, given a Petri net or an event structure a special transition system can be extracted, on which minimisation is performed. For instance in [13] the authors propose an encoding of safe Petri nets into a causal automata, in a way which preserves hp-bisimilarity. The causal automata can be transformed into a standard labelled transition system (LTS). In this way, the LTS representation can be used to check the equivalence between a pair of processes or to find a minimal representation of the behaviour. However, once a Petri net has been transformed into a causal automaton, then it is not possible to obtain the Petri net representation back, which can be of interest in some specific applications. In [9], the author uses a state transition diagram referred to as process graph, for the representation of the behaviour of a Petri net. Again, the transition diagram could be minimised with some technique for LTSs with structured states, but no direct approach is proposed.

Acknowledgements

We are grateful to the anonymous reviewers for their insightful suggestions which helped us improve the paper. This research was supported by the European Social Fund via the Doctoral Studies and Internationalisation Programme (DoRa) and by an institutional grant of the Estonian Research Council.

References

[1] M. Nielsen, G.D. Plotkin, G. Winskel, Petri nets, event structures and domains, part I, Theor. Comput. Sci. 13 (1) (1981) 85–108, http://dx.doi.org/10.1016/0304-3975(81)90112-2.
[2] G. Winskel, Event structures, in: W. Brauer, W. Reisig, G. Rozenberg (Eds.), Petri Nets: Applications and Relationships to Other Models of Concurrency, in: Lecture Notes in Computer Science, vol. 255, Springer, 1987, pp. 325–392.
[3] P. Baldan, A. Corradini, U. Montanari, Contextual Petri Nets, asymmetric event structures, and processes, Inf. Comput. 171 (1) (2001) 1–49, http://dx.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. Bakker, W.-P. Roever, G. Rozenberg (Eds.), Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency, School/Workshop, vol. 354, Springer, 1989, pp. 411–427.
[5] A.M. Rabinovich, B.A. Trakhtenbrot, Behaviour structures and nets, Fundam. Inform. 11 (4) (1988) 357–404.
[6] R. van Glabbeek, U. Goltz, Equivalence notions for concurrent systems and refinement of actions, in: A. Kreczmar, G. Mirkowska (Eds.), Mathematical Foundations of Computer Science, in: Lecture Notes in Computer Science, vol. 379, Springer, 1989, pp. 237–248.
[7] E. Best, R. Devillers, A. Kiehn, L. Pomello, Concurrent bisimulations in Petri nets, Acta Inform. 28 (3) (1991) 231–264, http://dx.doi.org/10.1007/BF01178506.
[8] A. Rensink, Posets for configurations! in: W.R. Cleaveland (Ed.), Proceedings of the 3rd International Conference on Concurrency Theory, in: Lecture Notes in Computer Science, vol. 630, Springer, 1992, pp. 269–285.
[9] R. van Glabbeek, History preserving process graphs, Draft, Available at http://theory.stanford.edu/~rvg/abstracts.html#hppg, 1996.
[10] R.J. van Glabbeek, G.D. Plotkin, Configuration structures, in: D. Kozen (Ed.), Proceedings of the Tenth Annual IEEE Symposium on Logic in Computer Science, IEEE Computer Society Press, 1995, pp. 199–209.
[11] I. Castellani, G.-Q. Zhang, Parallel product of event structures, Theor. Comput. Sci. 179 (1–2) (1997) 203–215, http://dx.doi.org/10.1016/S0304-3975(96)00104-1.
[12] A. Armas-Cervantes, P. Baldan, M. Dumas, L. García-Bañuelos, Behavioral comparison of process models based on canonically reduced event structures, in: S. Wasim Sadiq, P. Soffer, H. Völzer (Eds.), Proceedings of the 12th International Conference on Business Process Management, in: Lecture Notes in Computer Science, vol. 8659, Springer, 2014, pp. 267–282.
[13] U. Montanari, M. Pistore, Minimal transition systems for history-preserving bisimulation, in: R. Reischuk, M. Morvan (Eds.), Proceedings of the 14th Annual Symposium on Theoretical Aspects of Computer Science, in: Lecture Notes in Computer Science, vol. 1200, Springer, 1997, pp. 413–425.