Contextual Petri Nets, Asymmetric Event Structures, and Processes*

Paolo Baldan, Andrea Corradini, Ugo Montanari

Dipartimento di Informatica
University of Pisa
Corso Italia, 40, 56125 Pisa, Italy

E-mail: {baldan, andrea, ugo}@di.unipi.it

Abstract. We present an event structure semantics for *contextual nets*, an extension of P/T Petri nets where transitions can check for the presence of tokens without consuming them (read-only operations). A basic rôle is played by *asymmetric event structures*, a generalization of Winskel's prime event structures where symmetric conflict is replaced by a relation modelling *asymmetric conflict* or *weak causality*, used to represent a new kind of dependency between events arising in contextual nets. Extending Winskel's seminal work on safe nets, the truly concurrent event based semantics of contextual nets is given at categorical level via a chain of coreflections leading from the category **SW-CN** of semi-weighted contextual nets to the category **Dom** of finitary prime algebraic domains. First an unfolding construction generates from a contextual net a corresponding *occurrence contextual net*, from where an asymmetric event structure is extracted. Then the configurations of the asymmetric event structure, endowed with a suitable order, are shown to form a finitary prime algebraic domain.
We also investigate the relation between the proposed unfolding semantics and several deterministic process semantics for contextual nets in the literature. In particular, the domain obtained via the unfolding is characterized as the collection of the deterministic processes of the net endowed with a kind of prefix ordering.

1 Introduction

Petri nets are widely accepted as an adequate formalism for the specification of the behaviour of concurrent systems [30, 33]. In fact the state of a net has an intrinsic distributed nature, being a set of *tokens* distributed among a set of *places*. A *transition* is enabled in a state if enough tokens are present in the places in its precondition. The firing of the transition *removes* such tokens and *produces* new tokens in its postcondition. More transitions can fire together if they consume mutually disjoint sets of tokens. This makes clear that nets can specify in a natural way phenomena like mutual exclusion, concurrency, sequential composition and nondeterminism.

A limit in the expressiveness of Petri nets is represented by the fact that a transition can only consume and produce tokens, and thus a net cannot express in a natural way non-destructive reading operations. The naïve technique of representing the reading of a token via a consume/produce cycle causes a loss in concurrency. Consider the net N_0 in Figure 1, where the token in place s is intended to represent a resource which is accessed by two transitions t_0 and t_1 in a read-only modality. Differently from what one could expect the two transitions cannot read the shared resource s concurrently, but their accesses must be serialized.

Contextual nets. *Contextual nets* [28], also called nets with test arcs in [10], activator arcs in [21] or read arcs in [35], extend classical nets with the possibility of checking for the presence of tokens which are not consumed. Concretely, besides the usual preconditions and postconditions, a transition of a contextual

* Research partially supported by MURST project Tecniche Formali per Sistemi Software, by TMR Network GETGRATS and by Esprit WG APPLIGRAPH.

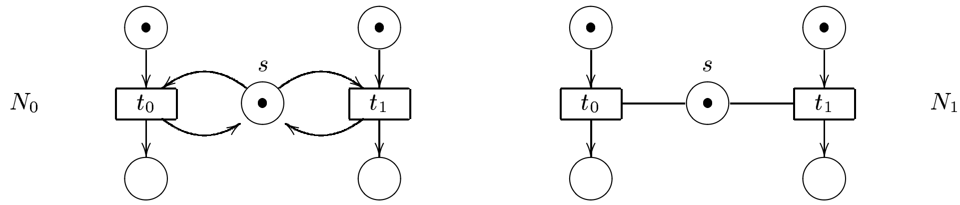


Fig. 1. Traditional nets do not allow for concurrent read-only operations.

net has also some *context* conditions, that, informally speaking, specify that the presence of some tokens in certain places is necessary for the transition to be fired, but such tokens are not affected by the firing of the transition. In other words, a context can be thought of as an item which is *read but not consumed* by the transition, in the same way as preconditions can be considered being read and consumed and postconditions being simply written. Coherently with this view, the same token can be used as context by many transitions at the same time and with multiplicity greater than one by the same transition. For instance, the situation of two agents reading a shared resource discussed above can be modelled directly by the contextual net N_1 of Figure 1, where the transitions t_0 and t_1 use the place s as context. According to the informal description of the semantics of contextual nets, in N_1 the transitions t_0 and t_1 can fire concurrently. Notice that in the pictorial representation of a contextual net directed arcs represent, as usual, preconditions and postconditions, while, following [28], non-directed (usually horizontal) arcs are used to represent context conditions.

The ability of faithfully representing the "reading of resources" allows contextual nets to model many concrete situations more naturally than classical nets. In recent years they have been used to model concurrent access to shared data (e.g., reading in a database) [34], to provide concurrent semantics to concurrent constraint (CC) programs [27], to model priorities [19] and to specify a net semantics for the π-calculus [8]. Moreover they have been studied for their connections with another powerful formalism for the specification of concurrent computations, namely graph transformation systems [28, 11]. If we think of the states of a net as sets (of tokens) labelled by place names, then a P/T net can be seen as a rewriting system on labelled sets (or equivalently on discrete graphs), the rewriting rules being specified by the transitions. Therefore contextual nets can be seen as an intermediate step between classical nets and graph grammars, and as such they can be used for transferring to graph grammars the great number of results and notions developed for nets (see e.g. [12, 1]).

In its seminal work [39], Winskel, starting from some results in [29], shows that the event structure semantics of *safe* nets can be given via a chain of coreflections starting from the category **Safe** of safe nets, through category **Occ** of occurrence nets. In particular, the event structure associated with a net is obtained by first constructing a "nondeterministic unfolding" of the net, and then by extracting from it the events (which correspond to transition occurrences) and the causal and conflict relations among them. In [24, 26] it has been shown that essentially the same construction applies to the wider category of *semi-weighted* nets, i.e., P/T nets in which the initial marking is a set and transitions can generate at most one token in each post-condition. It is worth noting that, besides being more general than safe nets, semi-weighted nets present the advantage of being characterized by a "static condition", not involving the behaviour but just the structure of the net. Figure 2 shows an example of semi-weighted P/T net, which is not safe.

This paper generalizes such result to the setting of *contextual nets* by showing that an event structure for a semi-weighted contextual net,[1] describing the behaviour of the net, can be obtained via a similar chain of coreflections. The resulting semantics is then shown to be "consistent" with the deterministic process semantics proposed in the literature for contextual nets.

We try next to outline the main problems which arose in such a development and the way we decided to solve them.

[1] Semi-weighted nets were called "weakly-safe nets" in the conference version of this paper [3].

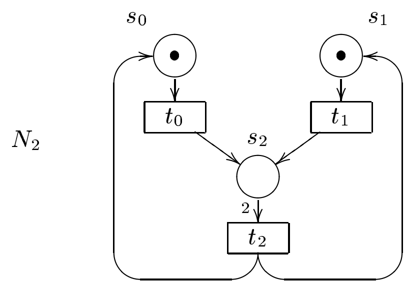


Fig. 2. A semi-weighted P/T net, which is not safe.

Asymmetric conflicts and asymmetric event structures. *Prime event structures* (PES) are a simple event based model of (concurrent) computations in which events are considered as atomic, indivisible and instantaneous steps, which can appear only once in a computation. An event can occur only after some other events (its causes) have taken place and the execution of an event can inhibit the execution of other events. This is formalized via two binary relations: *causality*, modelled by a partial order relation and *conflict*, modelled by a symmetric and irreflexive relation, hereditary with respect to causality.

When working with contextual nets the main critical point is the fact that the presence of context conditions leads to *asymmetric conflicts* or *weak dependencies* between events. To understand this basic concept, consider the net N_3 of Figure 3.(a), with two transitions t_0 and t_1 such that the same place s is a context for t_0 and a precondition for t_1.

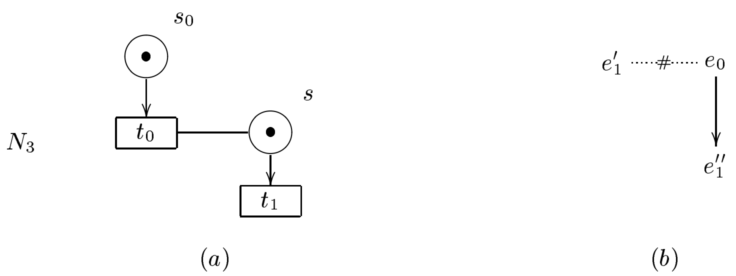


Fig. 3. A simple contextual net and a prime event structure representing its behaviour.

The possible firing sequences are t_0, t_1 and $t_0; t_1$, while $t_1; t_0$ is not allowed. Also the concurrent firing of t_0 and t_1 is not possible, differently from what happens in [21] and [35], the idea being that two concurrent events should be allowed to fire also in any order. This situation cannot be modelled in a direct way with a traditional prime event structure: t_0 and t_1 are neither in conflict nor concurrent nor causal dependent. Simply, as for a traditional conflict, the firing of t_1 prevents t_0 to be executed, so that t_0 can never follow t_1 in a computation. But the converse is not true, since t_1 *can* fire after t_0. This situation can be interpreted naturally as an *asymmetric conflict* between the two transitions. Equivalently, since t_0 precedes t_1 in any computation where both transitions are executed, in such computations t_0 acts as a cause of t_1. However, differently from a true cause, t_0 is not necessary for t_1 to be fired. Therefore we can also think of the relation between the two transitions as a *weak* form of *causal dependency*.

A reasonable way to encode this situation in a PES is to represent the firing of t_0 with an event e_0 and the firing of t_1 with two distinct mutually exclusive events: e_1', representing the execution of t_1 that prevents t_0, thus mutually exclusive with e_0; and e_1'', representing the execution of t_1 after t_0 (thus caused by e_0). Such PES is depicted in Figure 3.(b), where causal dependency is represented by a plain arrow and conflict is represented by a dotted line, labelled by #. However, this solution is not completely satisfactory with respect to the interpretation of contexts as "read-only resources": since t_0 just reads the token in s without changing it, one would expect the firing of t_1, preceded or not by t_0, to be represented by a unique event. This encoding may lead to an explosion of the size of the PES, since whenever an event is "duplicated" also all its consequences are duplicated. In addition it should be noted that it totally looses information on the new kind of dependency determined by read-only operations (which is "confused" with causality or symmetric conflict).

It is worth noting that the inability of representing the asymmetric conflict between events, without resorting to duplications is not specific to prime event structures, but it is basically related to the axiom of general Winskel's event structures (see [39], Definition 1.1.1) stating that the enabling relation $\vdash$ is "monotone" with respect to set inclusion:

$$A \vdash e \ \wedge \ A \subseteq B \ \wedge \ B \ \mathit{consistent} \quad \Rightarrow \quad B \vdash e.$$

As a consequence, the computational order between configurations is set inclusion, the idea being that if A and B are finite configurations such as $A \subseteq B$, then starting from A we can reach B by performing the events in $B - A$, whenever they become enabled. Obviously, this axiom does not hold in the presence of asymmetric conflict.

In order to provide a more direct, event based representation of contextual nets we introduce a new kind of event structure, called *asymmetric event structure* (AES). An AES, besides of the usual causal relation $\leq$ of a traditional prime event structure, has a relation $\nearrow$, that allows us to specify the new kind of dependency described above for transitions t_0 and t_1 of the net in Figure 3 simply as $t_0 \nearrow t_1$. As just remarked, the same relation has two natural interpretations: it can be thought of as an asymmetric version of conflict or as a weak form of causality. We decided to call it *asymmetric conflict*, but the reader should keep in mind both views, since in some situations it will be preferable to refer to the *weak causality* interpretation. Informally, in an AES each event has a set of "strong" causes (given by the causal dependency relation) and a set of weak causes (due to the presence of the asymmetric conflict relation). To be fired, each event must be preceded by all strong causes and by a (suitable) subset of the weak causes. Therefore, differently from PES's, the firing of an event can have more than one history. Moreover the usual symmetric binary conflict can be represented easily by using cycles of asymmetric conflict: for instance, if $e \nearrow e'$ and $e' \nearrow e$ then clearly e and e' can never occur in the same computation, since each one should precede the other.

Configurations of an AES are defined as sets of events representing possible computations of the AES. Then the set of configurations of an AES, ordered in a suitable way using the asymmetric conflict relation, turns out to be a finitary prime algebraic domain. The main difference with respect to the definition for classical event structures is that the order on configurations is not simply set-inclusion, essentially because a configuration C can not be extended with an event inhibited by an event already present in C. Such a construction extends to a functor from the category **AES** of asymmetric event structures to the category **Dom** of finitary prime algebraic domain, that establishes a coreflection between **AES** and **Dom**. By using the equivalence between the category **Dom** and the category **PES** of prime event structures [39] we can then recover a semantics in terms of traditional event structures. Essentially the PES semantics obtained in this way encodes the asymmetric conflict by using causality and symmetric conflict, as depicted in Figure 3. Therefore the AES semantics is finer than PES semantics since different AES's may be mapped to the same PES. It is remarkable that the "translation" from AES to PES semantics is done at categorical level, via a coreflection.

Several authors pointed out the inadequacy of Winskel's event structures for faithfully modeling general concurrent computations and they proposed alternative definitions. To model non deterministic choice, or equivalently, the possibility of having multiple disjunctive and mutually exclusive causes for an event, Boudol and Castellani [6] introduce the notion of *flow event structure*, where the causality relation is replaced by an irreflexive (in general non transitive) *flow relation*, representing essentially immediate causal dependency, and conflict is no more hereditary. To face a similar problem, Langerak [22] defines *bundle event structures*, where a set of multiple disjunctive and mutually exclusive causes for an event is called a *bundle set* for the event, and comes into play as a primitive notion. Asymmetric conflicts have been specifically treated by Pinna and Poigné in [31, 32], where the "operational" notion of event automaton suggests an enrichment of prime event structures and flow event structures with *possible causes*. The basic idea is that if e is a possible cause of e' then e can precede e' or it can be ignored, but the execution of e never follows e'. This is formalized by introducing an explicit subset of possible events in prime event structures or adding a "possible flow relation" in flow event structures. Similar ideas are developed, under a different perspective by Degano, Vigna and Gorrieri, in [14], where prioritized event structures are introduced as PES enriched with a partial order relation modeling priorities between events. Also bundle event structures have been extended by Langerak in [23] to take into account asymmetric conflicts.

Despite some differences in the definition and in the related notions, our AES can be seen as a generalization of event structures with possible events. On the other hand, flow event structures with possible flow and bundle event structures with asymmetric conflict would have been expressive enough for our aims, but less manageable than asymmetric event structures. For example, due to the presence of disjunctive causes, given an event there does not exist, in general, a least configuration which the event belongs to, and the problem of establishing if an event is executable in some computation becomes undecidable. Understanding which part of the results presented in this paper for AES's extends to flow event structures with possible flow and to bundle event structures with asymmetric conflict is an interesting matter of further investigation.

Unfolding for contextual nets. As for classical nets, the event structure semantics for a contextual net is obtained by first unfolding the net into an acyclic branching structure that is itself a contextual net. More precisely, an *unfolding* construction is presented which allows us to associate to each semi-weighted contextual net N an *occurrence* contextual net $\mathcal{U}_a(N)$ that describes in a static way the behaviour of N, by expressing possible events and the dependency relations between them. Each transition in $\mathcal{U}_a(N)$ represents a specific firing of a transition in N, and places in $\mathcal{U}_a(N)$ represent occurrences of tokens in the places of N. The unfolding operation can be extended to a functor $\mathcal{U}_a$ from **SW-CN** to the category **O-CN** of occurrence contextual nets, that is right adjoint to the inclusion functor $\mathcal{I}_O$: **O-CN** $\to$ **SW-CN**.

Transitions of an occurrence contextual net are related by causal dependency and asymmetric conflict, which are defined according to the previous discussion. Mutual exclusion is a derived relation, defined in terms of cycles of the asymmetric conflict relation. Thus, the semantics of semi-weighted contextual nets given in terms of occurrence contextual nets can be naturally abstracted to an AES semantics: given an occurrence contextual net we obtain an AES simply forgetting the places, but remembering the dependency relations that they induce between transitions. Again, this construction extends, at categorical level, to a coreflection between **AES** and **O-CN**. Therefore occurrence contextual nets can be seen as a convenient concrete representation of AES's, in the same way as occurrence nets represent PES [39] and flow nets represent flow event structures [5]. Finally, the coreflection between **AES** and **Dom**, discussed above, can be exploited to complete the chain of coreflections from **SW-CN** to **Dom**.

Independently from the conference version of this work, appeared as [3], an unfolding construction for contextual nets has been proposed by Vogler, Semenov and Yakovlev in [37]. The paper presents an algorithm for the construction of a finite prefix of the unfolding and proposes possible applications to the analysis of logic circuits. The unfolding construction, although developed only for finite safe contextual nets, is based on ideas analogous to ours and it leads, in that restricted case, to the same result. The main difference resides in the fact that Vogler et al. do not provide a categorical construction, but they develop their construction only at "object level". The advantages of having a categorical semantics defined via an adjunction are numerous. First, one is lead to consider a notion of morphism between systems (typically formalizing the idea of "simulation") and to define the semantical transformation consistently with such notion: a morphism between two systems must correspond to a morphism between their models. Moreover, there is often an obvious functor that maps models back into the category of systems (this is the case for nets, where occurrence contextual nets are particular contextual nets and thus such functor is simply the inclusion). Consequently the semantics can be defined naturally as the functor in the opposite direction, forming an adjunction, which (if it exists) is unique up to natural isomorphism. In other words, once one has decided the notion of simulation, there is a unique way to define the semantics consistently with such notion. Finally, several operations on nets (systems) may be expressed at categorical level as limit/colimit constructions. For instance, a pushout construction can be used to compose two nets, merging some part of them, obtaining a kind of generalized non-deterministic composition, while synchronization of nets can be modeled as a product (see [39, 26]). Since left/right adjoint functors preserve colimits/limits, a semantics defined via an adjunction turns out to be compositional with respect to such operations. The reader is referred to Goguen's paper [18] for an extensive discussion on the usefulness of category theory in computer science.

Relation with deterministic processes. The problem of providing a truly concurrent semantics for contextual nets based on (deterministic) processes has been faced by various authors (see, e.g., [28, 17, 19, 9, 36, 38]). Each deterministic process of a contextual net records the events occurring in a *single* computation of the net and the relationships existing between such events. Clearly, since the unfolding of a net is essentially a nondeterministic process that completely describes the behaviour of the net, one would expect that a relation could be established between the unfolding and the deterministic process semantics. Indeed, we show that, as already known for traditional nets [25], the domain associated to a semi-weighted contextual net N through the unfolding construction is isomorphic to the set of deterministic processes of the net starting from the initial marking, endowed with a kind of prefix ordering. This result is stated in an elegant categorical way. First a category $\mathbf{CP}[N]$ of concatenable processes for the net N is introduced, where objects are markings (states of the net), arrows are (decorated) processes (computations of the net) and arrow composition is an operation of concatenation of processes consistent with causal dependencies, modelling sequential composition of computations [17, 38]. Then the comma category $(m \downarrow \mathbf{CP}[N])$, where m is the initial marking of the net, is shown to be a preorder, inducing a partial order whose ideal completion is isomorphic to the domain associated to the unfolding. The proof relies on the categorical characterization of the unfolding, and in particular on the fact that, since the unfolding functor from **SW-CN** to **O-CN** is right adjoint to the inclusion, the counit of the adjunction immediately provides a one-to-one correspondence between the deterministic processes of a net N and those of its unfolding $\mathcal{U}_a(N)$.

Structure of the paper. The rest of the paper is organized as follows. Section 2 introduces the category **AES** of asymmetric event structures and describes some properties of such structures. Section 3 defines the coreflection between **AES** and the category **Dom** of finitary prime algebraic domains. Section 4 presents contextual nets and focuses on the category **SW-CN** of (semi-weighted) contextual nets. Section 5 is devoted to the definition and analysis of the category **O-CN** of occurrence contextual nets. Section 6 describes the unfolding construction for semi-weighted contextual nets and shows how such a construction gives rise, at categorical level, to a coreflection between **SW-CN** and **O-CN**. Section 7 completes the chain of coreflections from **SW-CN** to **Dom**, by presenting a coreflection between **O-CN** and **AES**. Section 8 discusses the relation between the unfolding and the deterministic process semantics of contextual nets. Finally Section 9 draws some conclusions and suggests possible directions for further research. An extended abstract of Sections 2-7 appeared in [3], while Section 8 is original.

2 Asymmetric event structures

We stressed in the introduction that PES's (and in general Winskel's event structures) are not expressive enough to model in a direct way the behaviour of models of computation, such as string, term, graph rewriting and contextual nets, where a rule may preserve a part of the state, in the sense that part of the state is necessary for the application of the rule, but it is not affected by such application.

To allow for a faithful description of the dependencies existing between events in such models, and in particular in contextual nets, this section introduces the category **AES** of *asymmetric event structures*, an extension of Winskel's prime event structures where the usual symmetric conflict relation is replaced by the new binary relation $\nearrow$, called *asymmetric conflict*. The intuition underlying the asymmetric conflict relation has been discussed in the introduction: if $e_0 \nearrow e_1$ then the firing of e_1 inhibits e_0, namely the execution of e_0 may precede the execution of e_1 or e_0 can be ignored, but e_0 cannot follow e_1. We will see that in this setting the symmetric binary conflict is no more a primitive relation, but it is represented via "cycles" of asymmetric conflict. For instance, if $e \nearrow e'$ and $e' \nearrow e$ then clearly e and e' can never occur in the same computation. As a consequence, PES's can be identified with a special subclass of asymmetric event structures, namely those where all conflicts are actually symmetric.

Let us start by introducing some basic notations on sets, relations and functions.

Notation 1 *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., $r \cap (Y \times Y)$;*
- *r^+ denotes the transitive closure of r, and r^* denotes the reflexive and transitive closure of r;*
- *r is* well-founded *if it has no infinite descending chains, i.e., $\langle e_i \rangle_{i \in \mathbb{N}} \in X$ 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 it has no (non-trivial) cycles;*
- *r is called a* preorder *if it is reflexive and transitive; it is a* partial order *if it is also antisymmetric.*

If $f : X \to X'$ is a partial function and $x \in X$, we write $f(x) = \bot$ to denote that f is not defined on x. Finally, the powerset of a set X is denoted by $\mathbf{2}^X$, while $\mathbf{2}^X_{fin}$ denotes the set of finite subsets of X.

It is worth recalling the formal definition of the category **PES** of prime event structures with binary conflicts, informally described in the introduction.

Definition 1 (prime event structures). *A* prime event structure (PES) *is a tuple $P = \langle E, \leq, \# \rangle$, where E is a set of* events *and $\leq$, $\#$ are binary relations on E called* causality relation *and* conflict relation *respectively, such that:*

1. *the relation $\leq$ is a partial order and $\lfloor e \rfloor = \{e' \in E : e' \leq e\}$ is finite for all $e \in E$;*
2. *the relation $\#$ is irreflexive, symmetric and hereditary with respect to $\leq$, i.e., $e \# e' \leq e''$ implies $e \# e''$ for all $e, e', e'' \in E$;*

Let $P_0 = \langle E_0, \leq_0, \#_0 \rangle$ and $P_1 = \langle E_1, \leq_1, \#_1 \rangle$ be two PES's. A PES-morphism *$f : P_0 \to P_1$ is a partial function $f : E_0 \to E_1$ such that:*

1. *for all $e_0 \in E_0$, if $f(e_0)$ is defined then $\lfloor f(e_0) \rfloor \subseteq f(\lfloor e_0 \rfloor)$;*
2. *for all $e_0, e_0' \in E_0$*
   (a) $f(e_0) \#_1 f(e_0') \quad \Rightarrow \quad e_0 \#_0 e_0'$;
   (b) $(f(e_0) = f(e_0')) \wedge (e_0 \neq e_0') \quad \Rightarrow \quad e_0 \#_0 e_0'$;

The category of prime event structures and PES-morphisms is denoted by **PES**.

We can now define the notion of asymmetric event structure. The basic ideas for the treatment of asymmetric conflict in our approach are similar to those suggested by Pinna and Poigné in [31,32]. In such papers they concentrate on event automata and on the distinction between specifications (given in the form of event structures) and automata implementing such specifications. Moreover, looking for event structures that allow to specify adequately features like priority and asymmetric conflict, they introduce the idea of possible events, namely events that, according to the considered computation, may be part or not of the causes of other events. Consequently the notion of PES with possible events and that of flow event structure with possible flow are considered. Apart from a different presentation, asymmetric event structures can be seen as a generalization of PES with possible events. Using their terminology, when $e_0 \nearrow e_1$ we can say that e_0 is a possible cause of e_1. However, differently from what happens for event structures with possible events, where a distinct set of possible events is singled out, our notion of possible cause is local, being induced by the asymmetric conflict relation. The extended bundle event structures of Langerak [23] share with our approach, besides the above mentioned basic ideas, the intuition that when asymmetric conflict is available, the symmetric conflict becomes useless, since it can be represented as an asymmetric conflict in both directions.

Definition 2 (asymmetric event structure). *An* asymmetric event structure (AES) *is a tuple $G = \langle E, \leq, \nearrow \rangle$, where E is a set of* events *and $\leq$, $\nearrow$ are binary relations on E called* causality relation *and* asymmetric conflict *respectively, such that:*

1. *the relation $\leq$ is a partial order and $\lfloor e \rfloor = \{e' \in E : e' \leq e\}$ is finite for all $e \in E$;*
2. *the relation $\nearrow$ satisfies, for all $e, e' \in E$:*
   (a) $e < e' \quad \Rightarrow \quad e \nearrow e'$;[2]

[2] With $e < e'$ we mean $e \leq e$ and $e \neq e'$.

(b) $\nearrow_{\lfloor e \rfloor}$ *is acyclic.*[3]

If $e \nearrow e'$*, according to the double interpretation of* $\nearrow$ *we say that* e is prevented *by* e' *or* e weakly causes e'*. Moreover we say that* e is strictly prevented *by* e' *(or* e strictly weakly causes e'*), written* $e \leadsto_G e'$*, if* $e \nearrow e'$ *and* $\neg(e < e')$*.*

The definition can be easily understood by giving a more formal account of the ideas presented in the introduction. Let $\mathit{Fired}_C(e)$ denote the fact that the event e has been fired in a computation C and let $\mathit{prec}_C(e, e')$ denote that e precedes e' in the computation. Then

$$e < e' \quad \stackrel{def}{\equiv} \quad \forall C.\ \mathit{Fired}_C(e') \ \Rightarrow\ \mathit{Fired}_C(e) \ \wedge\ \mathit{prec}_C(e, e')$$
$$e \nearrow e' \quad \stackrel{def}{\equiv} \quad \forall C.\ \mathit{Fired}_C(e) \wedge \mathit{Fired}_C(e') \ \Rightarrow\ \mathit{prec}_C(e, e')$$

Therefore $<$ represents a global order of execution, while $\nearrow$ determines an order of execution only locally to each computation. Thus it is natural to impose $\nearrow$ to be an extension of $<$. Moreover notice that if a set of events form an asymmetric conflict cycle then such events cannot appear in the same computation, otherwise the execution of each event should precede the execution of the event itself. This explains why we require the transitive closure of $\nearrow$, restricted to the causes $\lfloor e \rfloor$ of an event e, to be acyclic (and thus well-founded, being $\lfloor e \rfloor$ finite). Otherwise not all causes of e could be executed in the same computation and thus e itself could not be executed. The informal interpretation makes also clear that $\nearrow$ is *not* in general transitive. If $e \nearrow e' \nearrow e''$ it is not true that e must precede e'' when both fire. This holds only in a computation where also e' fires.

The fact that a set of n events in an asymmetric conflict cycle can never occur in the same computation can be naturally interpreted as a form of n-ary conflict. More formally, it is useful to associate to each AES an explicit conflict relation (on sets of events) defined in the following way:

Definition 3 (induced conflict relation). *Let* $G = \langle E, \leq, \nearrow \rangle$ *be an* AES*. The* conflict *relation* $\#^a \subseteq \mathbf{2}^E_{\mathit{fin}}$ *associated to* G *is defined as:*

$$\frac{e_0 \nearrow e_1 \nearrow \ldots \nearrow e_n \nearrow e_0}{\#^a\{e_0, e_1, \ldots, e_n\}} \qquad\qquad \frac{\#^a(A \cup \{e\}) \quad e \leq e'}{\#^a(A \cup \{e'\})}$$

where A *denotes a generic finite subset of* E*. The superscript* a *in* $\#^a$ *reminds that this relation is induced by asymmetric conflict. Sometimes we use the infix notation for the "binary version" of the conflict, i.e., we write* $e \#^a e'$ *for* $\#^a\{e, e'\}$*.*

Notice that if $\#^a A$ then $\lfloor A \rfloor$ contains a cycle of asymmetric conflict, and, viceversa, if $\lfloor A \rfloor$ contains a cycle $e_0 \nearrow e_1 \ldots e_n \nearrow e_0$ then there exists a subset $A' \subseteq A$ such that $\#^a A'$ (for instance, choosing an event $a_i \in A$ such that $e_i \leq a_i$ for $i \in \{0, \ldots, n\}$ then A' can be the set $\{a_i \mid i \in \{0, \ldots, n\}\}$).

The induced binary conflict $\#^a$ satisfies all the properties of the conflict relation of traditional PES's, i.e., it is irreflexive (since for each event e, by definition $(\nearrow_{\lfloor e \rfloor})^+$ is irreflexive), symmetric and hereditary with respect to $\leq$. Therefore $\langle E, \leq, \#^a \rangle$ is a PES, in the following referred to as the PES underlying the AES G.

The notion of AES-morphism is a quite natural extension of that of PES-morphism. Intuitively, it is a (possibly partial) mapping of events that "preserves computations".

Definition 4 (AES-morphism). *Let* $G_0 = \langle E_0, \leq_0, \nearrow_0 \rangle$ *and* $G_1 = \langle E_1, \leq_1, \nearrow_1 \rangle$ *be two* AES*'s. An* AES*-morphism* $f : G_0 \to G_1$ *is a partial function* $f : E_0 \to E_1$ *such that:*

1. *for all* $e_0 \in E_0$*, if* $f(e_0)$ *is defined then* $\lfloor f(e_0) \rfloor \subseteq f(\lfloor e_0 \rfloor)$*;*
2. *for all* $e_0, e_0' \in E_0$
   (a) $(f(e_0) = f(e_0')) \wedge (e_0 \neq e_0') \quad \Rightarrow \quad e_0 \#^a_0 e_0'$*;*
   (b) $f(e_0) \nearrow_1 f(e_0') \quad \Rightarrow \quad (e_0 \nearrow_0 e_0') \vee (e_0 \#^a_0 e_0')$*.*

[3] Equivalently, we can require $(\nearrow_{\lfloor e \rfloor})^+$ irreflexive. This implies that, in particular, $\nearrow$ is irreflexive.

Notation 2 *In the following when considering a* PES P *and an* AES G, *we implicitly assume that* $P = \langle E, \leq, \# \rangle$ *and* $G = \langle E, \leq, \nearrow \rangle$. *Moreover superscripts and subscripts on the structure name carry over the names of the involved sets, functions and relations (e.g.,* $G_i = \langle E_i, \leq_i, \nearrow_i \rangle$*).*

The next lemma states that AES-morphisms, analogously to what happens for PES's, reflect the (induced) conflict relation.

Lemma 1 (AES-morphisms reflect conflicts). *Let* G_0 *and* G_1 *be two* AES*'s and let* $f : G_0 \to G_1$ *be an* AES*-morphism. Given a set of events* $A \subseteq_{fin} E_0$, *if* $\#_1^a f(A)$ *then* $\#_0^a A'$ *for some* $A' \subseteq A$.

Proof. Let $A \subseteq_{fin} E_0$ and let $\#_1^a f(A)$. Therefore, by definition of conflict there exists a $\nearrow_1$-cycle $e'_0 \nearrow_1 e'_1 \nearrow_1 \dots \nearrow_1 e'_n \nearrow_1 e'_0$ in $\lfloor f(A) \rfloor$. By definition of AES-morphism, we have that $\lfloor f(A) \rfloor \subseteq f(\lfloor A \rfloor)$ and thus we can find $e_0, \dots, e_n \in \lfloor A \rfloor$ such that $e'_i = f(e_i)$ for all $i \in \{0, \dots, n\}$. Let us fix, for each $i \in \{0, \dots, n\}$, an event $a_i \in A$ such that $e_i \leq a_i$.

Now, if $e_i \#_0^a e_{i+1}$ for some i (we conventionally identify $n+1$ with 0) then the conflictual subset of A we are looking for is $\{a_i, a_{i+1}\}$. Otherwise, by definition of AES-morphism, we have $e_0 \nearrow_0 e_1 \nearrow_0 \dots \nearrow_0 e_0$, and thus $\#_0^a \{a_i \mid i \in \{0, \dots, n\}\}$. ◻

As an immediate consequence of the previous lemma, every AES-morphism is a PES-morphism between the underlying PES's. Moreover it can be easily shown that AES-morphisms are closed under composition.

Lemma 2 (composition). AES*-morphisms are closed under composition.*

Proof. Let $f_0 : G_0 \to G_1$ and $f_1 : G_1 \to G_2$ be two AES-morphisms and consider $f_1 \circ f_0 : G_0 \to G_2$. The proof that $f_1 \circ f_0$ satisfies conditions (1) and (2a) of Definition 4 is trivial and analogous to that for traditional PES's. As for condition (2b), let $e_0, e'_0 \in E_0$, with $f_1(f_0(e_0)) \nearrow_2 f_1(f_0(e'_0))$. Then, by definition of AES-morphism,

$$f_0(e_0) \nearrow_1 f_0(e'_0) \quad \text{or} \quad f_0(e_0) \#_1^a f_0(e_0).$$

In the first case, $e_0 \nearrow_0 e'_0$ or $e_0 \#_0^a e'_0$. In the second case, since AES-morphisms reflect conflicts, $e_0 \#_0^a e'_0$. Thus all AES-morphisms conditions are verified. ◻

Definition 5 (category AES). *We denote by* **AES** *the category having asymmetric event structures as objects and* AES*-morphisms as arrows.*

We conclude this section by formalizing the intuition that AES's are a proper extension of PES's. As one would expect, PES's can be identified with the subclass of AES's where the strict asymmetric conflict relation is actually symmetric.

Lemma 3. *Let* $P = \langle E, \leq, \# \rangle$ *be a* PES. *Then* $G = \langle E, \leq, < \cup \# \rangle$ *is an* AES, *where the asymmetric conflict relation is defined as the union of the "strict" causality and conflict relations.*

Moreover, if $f : P_0 \to P_1$ *is a* PES*-morphism then* f *is an* AES*-morphism between the corresponding* AES*'s* G_0 *and* G_1, *and if* $g : G_0 \to G_1$ *is an* AES*-morphism then it is also a* PES*-morphism between the original* PES*'s.*

Proof. Let $P = \langle E, \leq, \# \rangle$ be a PES. The fact that $G = \langle E, \leq, < \cup \# \rangle$ is an AES is a trivial consequence of the definitions. In particular, the asymmetric conflict relation of G is acyclic on the causes of each event since $\#$ is irreflexive and $<$ is a partial order in P.

Now, let $f : P_0 \to P_1$ be a PES-morphism. To prove that f is also an AES-morphism between the corresponding AES G_0 and G_1, first observe that the induced binary conflict relation of G is exactly $\#$, and thus Conditions (1) and (2a) are the same for PES- and AES-morphisms (see Definitions 1 and 4). As for Condition (2b), if $f(e_0) \nearrow f(e_1)$, then, by definition, $f(e_0) < f(e_1)$ or $f(e_0) \# f(e_1)$ and thus, by properties of PES's (easily derivable from Definition 1), in the first case $e_0 < e_1$ or $e_0 \# e_1$ whilst, in the second case, $e_0 \# e_1$.

Similar considerations allow to conclude that, if $g : G_0 \to G_1$ is an AES-morphism, then it is also a PES-morphism between the original PES's. ◻

By the previous lemma, there is a full embedding functor from **PES** into **AES** that replaces the symmetric conflict with an asymmetric conflict relation equal to the union of "strict" causality and conflict relation in each PES, and that is identity on arrows.

Proposition 1 (prime and asymmetric event structures). *The functor* $\mathcal{J} : \mathbf{PES} \to \mathbf{AES}$ *defined by*

- $\mathcal{J}(\langle E, \leq, \# \rangle) = \langle E, \leq, < \cup \# \rangle$*;*
- $\mathcal{J}(f : P_0 \to P_1) = f$*.*

is a full embedding of **PES** *into* **AES***.*

3 From asymmetric event structures to domains

Prime event structures are intimately connected to prime algebraic domains, another mathematical structure widely used in semantics. More precisely the category **PES** of prime event structures is equivalent to the category **Dom** of (finitary coherent) prime algebraic domains. For asymmetric event structures this result generalizes to the existence of a coreflection between **AES** and **Dom**. Such a coreflection allows for an elegant translation of an AES semantics into a domain, and thus into a classical PES semantics. The PES semantics obtained in this way represents asymmetric conflicts via symmetric conflict and causality with a duplication of events, as described in the introduction (see Figure 3).

3.1 Prime event structures and domains

This subsection reviews the definition of the category **Dom** of finitary prime algebraic domains and the equivalence between **Dom** and the category **PES** of prime event structures [39], which will be needed in the sequel.

First we need some basic notions and notations for partial orders. A preordered or partially ordered set $\langle D, \sqsubseteq \rangle$ will be often denoted simply as D, by omitting the (pre)order relation. Given an element $x \in D$, we write $\downarrow x$ to denote the set $\{y \in D \mid y \sqsubseteq x\}$. A subset $X \subseteq D$ is *compatible*, written $\uparrow X$, if there exists an upper bound $d \in D$ for X (i.e., $x \sqsubseteq d$ for all $x \in X$). It is *pairwise compatible* if $\uparrow \{x, y\}$ (often written $x \uparrow y$) for all $x, y \in X$. A subset $X \subseteq D$ is called *directed* if for any $x, y \in X$ there exists $z \in X$ such that $x \sqsubseteq z$ and $y \sqsubseteq z$.

Definition 6 ((finitary) (algebraic) complete partial order). *A partial order D is* (directed) complete (CPO) *if for any directed subset $X \subseteq D$ there exists the least upper bound $\bigsqcup X$ in D. An element $e \in D$ is* compact *if for any directed set $X \subseteq D$, $e \sqsubseteq \bigsqcup X$ implies $e \sqsubseteq x$ for some $x \in X$. The set of* compact *elements of D is denoted by $\mathsf{K}(D)$.*

A CPO *D is called* algebraic *if for any $x \in D$, $x = \bigsqcup(\downarrow x \cap \mathsf{K}(D))$. We say that D is* finitary *if for each compact element $e \in D$ the set $\downarrow e$ is finite.*

Given a finitary algebraic CPO D we can think of its elements as "pieces of information" expressing the states of evolution of a process. Finite elements represent states which are reached after a finite number of steps. Thus algebraicity essentially says that each infinite computation can be approximated with arbitrary precision by the finite ones.

Winskel's domains satisfy stronger completeness properties, which are formalized by the following definition.

Definition 7 ((prime algebraic) coherent poset). *A partial order D is called* coherent *(*pairwise complete*) if for all pairwise compatible $X \subseteq D$, there exists the least upper bound $\bigsqcup X$ of X in D.*

A complete prime *of D is an element $p \in D$ such that, for any compatible $X \subseteq D$, if $p \sqsubseteq \bigsqcup X$ then $p \sqsubseteq x$ for some $x \in X$. The set of complete primes of D is denoted by $Pr(D)$. The partial order D is called* prime algebraic *if for any element $d \in D$ we have $d = (\bigsqcup \downarrow d \cap Pr(D))$. The set $\downarrow d \cap Pr(D)$ of complete primes of D below d will be denoted $Pr(d)$.*

Being not expressible as the least upper bound of other elements, complete primes of D can be seen as elementary indivisible pieces of information (events). Thus prime algebraicity expresses the fact that all the possible computations of the system at hand can be obtained by composing these elementary blocks of information.

Notice that directed sets are pairwise compatible, and thus each coherent partial order is a CPO. For the same reason each complete prime is a compact element, namely $Pr(D) \subseteq \mathsf{K}(D)$ and thus prime algebraicity implies algebraicity. Moreover if D is coherent then for each non empty $X \subseteq D$ there exists the greatest lower bound $\bigsqcap X$, which can be expressed as $\bigsqcup\{y \in D \mid \forall x \in X.\ y \sqsubseteq x\}$.

Definition 8 (domains). *The partial orders we shall work with are coherent, prime algebraic, finitary partial orders, hereinafter simply referred to as* (Winskel's) domains.[4]

The definition of morphism between domains is based on the notion of immediate precedence. Given a domain D and two distinct elements $d \neq d' \in D$ we say that d is an *immediate predecessor* of d', written $d \prec d'$ if

$$d \sqsubseteq d' \ \wedge \forall d'' \in D.\ (d \sqsubseteq d'' \sqsubseteq d' \ \Rightarrow \ d'' = d \ \vee \ d'' = d')$$

Moreover we write $d \preceq d'$ if $d \prec d'$ or $d = d'$. According to the informal interpretation of domain elements sketched above, $d \preceq d'$ intuitively means that d' is obtained from d by adding a quantum of information. Domain morphisms are required to preserve such relation.

Definition 9 (category Dom). *Let D_0 and D_1 be domains. A* domain morphism $f : D_0 \to D_1$ *is a function, such that:*

- $\forall x, y \in D_0$, *if* $x \preceq y$ *then* $f(x) \preceq f(y)$. ($\preceq$-preserving)
- $\forall X \subseteq D_0$, X *pairwise compatible,* $f(\bigsqcup X) = \bigsqcup f(X)$; (Additive)
- $\forall X \subseteq D_0$, $X \neq \emptyset$ *and compatible,* $f(\bigsqcap X) = \bigsqcap f(X)$; (Stable)

We denote by **Dom** *the category having domains as objects and domain morphisms as arrows.*

The category **Dom** is shown in [39] to be equivalent to the category **PES**, the equivalence being established by two functors $\mathcal{L} : \mathbf{PES} \to \mathbf{Dom}$ and $\mathcal{P} : \mathbf{Dom} \to \mathbf{PES}$

$$\mathbf{PES} \underset{\mathcal{L}}{\overset{\mathcal{P}}{\underset{\sim}{\rightleftarrows}}} \mathbf{Dom}$$

The functor $\mathcal{L}$ associates to each PES the partial order of its configurations (subsets of events, left-closed with respect to causality and conflict free), ordered by subset inclusion. The image via $\mathcal{L}$ of a PES-morphism $f : P_0 \to P_1$ is the obvious extension of f to sets of events.

A more accurate description of the functor $\mathcal{P}$ is needed, since such functor will be used to map domains back into asymmetric event structures. A fundamental rôle is played by the notion of prime interval.

Definition 10 (prime interval). *Let $\langle D, \sqsubseteq \rangle$ be a domain. A* prime interval *is a pair $[d, d']$ of elements of D such that $d \prec d'$. Let us define*

$$[c, c'] \leq [d, d'] \quad \textit{if } (c = c' \sqcap d) \ \wedge \ (c' \sqcup d = d'),$$

and let $\sim$ be the equivalence obtained as the transitive and symmetric closure of (the preorder) $\leq$.

The intuition that a prime interval represents a pair of elements differing only for a "quantum" of information is confirmed by the fact that there exists a bijective correspondence between $\sim$-classes of prime intervals and complete primes of a domain D (see [29]). More precisely, the map

[4] The use of this kind of structures in semantics have been first investigated by Berry [4], where they are called *dI-domains*. The relation between Winskel domains and dI-domains, which are finitary distributive consistent-complete algebraic CPO's is established by the fact that for a finitary algebraic consistent-complete (or coherent) CPO, prime algebraicity is equivalent to distributivity.

$$[d, d']_\sim \mapsto p,$$

where p is the unique element in $Pr(d') - Pr(d)$, is an isomorphism between the $\sim$-classes of prime intervals of D and the complete primes $Pr(D)$ of D, whose inverse is the function:

$$p \mapsto [\bigsqcup\{c \in D \mid c \sqsubset p\}, p]_\sim.$$

The above machinery allows us to give the definition of the functor $\mathcal{P}$ "extracting" an event structure from a domain.

Definition 11 (from domains to PES's). *The functor* $\mathcal{P} : \mathbf{Dom} \to \mathbf{PES}$ *is defined as follows:*

- *given a domain* D, $\mathcal{P}(D) = \langle Pr(D), \leq, \# \rangle$ *where*

$$p \leq p' \quad \textit{iff} \quad p \sqsubseteq p' \qquad \textit{and} \qquad p \# p' \quad \textit{iff} \quad \neg(p \uparrow p');$$

- *given a domain morphism* $f : D_0 \to D_1$, *the morphism* $\mathcal{P}(f) : \mathcal{P}(D_0) \to \mathcal{P}(D_1)$ *is the function:*

$$\mathcal{P}(f)(p_0) = \begin{cases} p_1 \text{ if } p_0 \mapsto [d_0, d_0']_\sim,\ f(d_0) \prec f(d_0') \text{ and } [f(d_0), f(d_0')]_\sim \mapsto p_1; \\ \bot \text{ otherwise, i.e., if } f(d_0) = f(d_0'). \end{cases}$$

3.2 Asymmetric event structures and domains

This subsection defines a coreflection between the category **AES** of asymmetric event structures and the category **Dom** of domains. The domain associated to an AES G is obtained by considering the configurations of G, suitably ordered using the asymmetric conflict relation. Viceversa, given a domain D we obtain the corresponding AES by applying first the functor $\mathcal{P} : \mathbf{Dom} \to \mathbf{PES}$ and then the embedding $\mathcal{J} : \mathbf{PES} \to \mathbf{AES}$, defined in Proposition 1.

Generally speaking, a configuration of an event structure is a set of events representing a possible computation of the system modelled by the event structure. The presence of the asymmetric conflict relation makes such definition slightly more involved than the traditional one.

Definition 12 (configuration). *Let* $G = \langle E, \leq, \nearrow \rangle$ *be an* AES. *A* configuration *of* G *is a set of events* $C \subseteq E$ *such that*

1. $\nearrow_C$ *is well-founded;*
2. $\{e' \in C \mid e' \nearrow e\}$ *is finite for all* $e \in C$;
3. C *is left-closed with respect to* $\leq$, *i.e., for all* $e \in C$, $e' \in E$, $e' \leq e$ *implies* $e' \in C$.

The set of all configurations of G *is denoted by* $Conf(G)$.

Condition 1 first ensures that in C there are no $\nearrow$-cycles, and thus excludes the possibility of having in C a subset of events in conflict (formally, for any $A \subseteq_{fin} C$, we have $\neg(\#^a A)$). Moreover it guarantees that $\nearrow$ has no infinite descending chains in C, that, together with Condition 2, implies that the set $\{e' \in C \mid e' (\nearrow_C)^+ e\}$ is finite for each event e in C; thus each event has to be preceded only by finitely many other events of the configuration. Finally Condition 3 requires that all the causes of each event are present.

If a set of events A satisfies only the first two properties of Definition 12 it is called *consistent* and we write $co(A)$. Notice that, unlike for traditional event structures, consistency is not a finitary property.[5] For instance, let $A = \{e_i \mid i \in \mathbb{N}\} \subseteq E$ be a set of events such that all e_i's are distinct and $e_{i+1} \nearrow e_i$ for all $i \in \mathbb{N}$. Then A is not consistent, but each finite subset of A is.

Let us now define an order $\sqsubseteq$ on the configurations of an AES, aimed at formalizing the idea of "computational extension", namely such that $C_1 \sqsubseteq C_2$ if the configuration C_1 can evolve into C_2. A remarkable difference with respect to the classical approach is that the order on configurations is not simply set-inclusion, since a configuration C cannot be extended with an event inhibited by some of the events already present in C.

[5] A property Q on the subsets of a set X is *finitary* if given any $Y \subseteq X$, from $Q(Z)$ for all finite subsets $Z \subseteq Y$ it follows $Q(Y)$.

Definition 13 (extension). *Let $G = \langle E, \leq, \nearrow\rangle$ be an* AES *and let $A, A' \subseteq E$ be sets of events. We say that A'* extends *A and we write $A \sqsubseteq A'$, if*

1. *$A \subseteq A'$;*
2. *$\neg(e' \nearrow e) \quad$ for all $e \in A,\ e' \in A' - A$.*

Often in the sequel it will be preferable to use the following condition, equivalent to (2):

$$\forall e \in A.\ \forall e' \in A'.\ e' \nearrow e \ \Rightarrow\ e' \in A.$$

The extension relation is a partial order on the set $Conf(G)$ of configurations of an AES. Our aim is now to prove that $\langle Conf(G), \sqsubseteq\rangle$ is a finitary prime algebraic domain. This means that like prime event structures [39], flow event structure [5], and prioritized event structures [14], also asymmetric event structures provide a concrete presentation of prime algebraic domains.

Given an AES G, in the following we will denote by $Conf(G)$ both the set of configurations of G and the corresponding partial order. The following proposition presents a simple but useful property of the partial order of configurations of an AES, strictly connected with coherence.

Lemma 4. *Let G be an* AES *and let $A \subseteq Conf(E)$ be a pairwise compatible set of configurations. Then for all $C \in A$ and $e \in C$*

$$e' \in \bigcup A \ \wedge\ e' \nearrow e \quad \Rightarrow \quad e' \in C;$$

Proof. Let $e' \in \bigcup A$ be an event such that $e' \nearrow e$. Then there is a configuration $C' \in A$ such that $e' \in C'$. Being C and C' compatible, there is $C'' \in Conf(G)$ such that $C, C' \sqsubseteq C''$. Thus $e' \in C''$ and, since $C \sqsubseteq C''$, by definition of $\sqsubseteq$ we conclude that $e' \in C$. □

The next lemma proves that for pairwise compatible sets of configurations the least upper bound and the greatest lower bound are simply given by union and intersection.

Lemma 5 ($\bigsqcup$ and $\bigsqcap$ for sets of configurations). *Let G be an* AES*. Then*

1. *if $A \subseteq Conf(E)$ is pairwise compatible then $\bigsqcup A = \bigcup A$;*
2. *if $C_0 \uparrow C_1$ then $C_0 \sqcap C_1 = C_0 \cap C_1$.*

Proof. 1. Let $A \subseteq Conf(E)$ be a pairwise compatible set of configurations. First notice that $\bigcup A$ is a configuration. In fact:
- $\nearrow_{\bigcup A}$ is well-founded.
  Let us suppose that there is in $\bigcup A$ an infinite descending chain:
  $$\ldots e_{i+1} \nearrow e_i \nearrow e_{i-1} \nearrow \ldots \nearrow e_0.$$
  Let $C \in A$ such that $e_0 \in C$. Lemma 4, together with an inductive reasoning, ensure that this infinite chain is entirely contained in C. But this contradicts $C \in Conf(G)$.
- $\{e' \in \bigcup A \mid e' \nearrow e\}$ is finite for all $e \in \bigcup A$.
  Let $e \in \bigcup A$, then there exists $C \in A$ such that $e \in C$. By Lemma 4, the set $\{e' \in \bigcup A \mid e' \nearrow e\} = \{e' \in C \mid e' \nearrow e\}$, and thus it is finite.
- $\bigcup A$ is left-closed, since each $C \in A$ is left-closed.

The configuration $\bigcup A$ is an upper bound for A. In fact, for any $C \in A$, clearly $C \subseteq \bigcup A$ and for all $e \in C$, $e' \in \bigcup A$, if $e' \nearrow e$ then, by Lemma 4, $e' \in C$. Thus $C \sqsubseteq \bigcup A$. Moreover, if C_0 is another upper bound for A, namely a configuration such that $C \sqsubseteq C_0$ for all $C \in A$, then $\bigcup A \subseteq C_0$. Furthermore for any $e \in \bigcup A$, $e' \in C_0$ with $e' \nearrow e$, since $e \in C$ for some $C \in A$ we conclude that $e' \in C \subseteq \bigcup A$. Thus $\bigcup A \sqsubseteq C_0$ and this shows that $\bigcup A$ is the least upper bound of A.

2. Let $C_0 \uparrow C_1$ be two compatible configurations and let $C = C_0 \cap C_1$. Then it is easily seen that C is a configuration. Moreover $C \sqsubseteq C_0$. In fact $C \subseteq C_0$ and for all $e \in C$, $e' \in C_0$, if $e' \nearrow e$, since $e \in C_1$ and $C_0 \uparrow C_1$, by Lemma 4, $e' \in C_1$ and thus $e' \in C$. In the same way $C \sqsubseteq C_1$, and thus C is a lower bound for C_0 and C_1. To show that C is the greatest lower bound observe that if $C' \sqsubseteq C_0, C_1$ is another lower bound then, clearly $C' \subseteq C$. Furthermore, if $e \in C'$, $e' \in C$ with $e' \nearrow e$, since, in particular, $e' \in C_0$, we conclude $e' \in C'$. Hence $C' \sqsubseteq C$. □

For traditional event structures an event e uniquely determines its history, that is the set $\lfloor e \rfloor$ of its causes, independently of the configuration at hand. In the case of asymmetric event structures, instead, an event e may have different histories, in the sense that the set of events that must precede e in a configuration C depends on C. Essentially, the possible histories of e are obtained inserting or not in a configuration the weak causes of e, that thus can be seen as "possible causes".

Definition 14 (possible history). *Let G be an* AES *and let $e \in E$. Given a configuration $C \in Conf(G)$ 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*

$$Hist(e) = \{C[\![e]\!] \mid C \in Conf(E) \ \wedge \ e \in C\}.$$

We denote by $Hist(G)$ the set of possible histories of all events in G, namely

$$Hist(G) = \bigcup\{Hist(e) \mid e \in E\}.$$

Notice that each history is a *finite* set of events since $\nearrow_C$ is well founded for each configuration C.

Let us now give some properties of the set of histories. Point 1 shows that each history of an event e in a configuration C, is itself a configuration which is extended by C. Point 2 essentially states that although an event e has in general more than one history, as one would expect, the history can not change after the event has occurred. Point 3 asserts that different histories of the same event are incompatible.

Lemma 6 (history properties). *Let G be an* AES. *Then in $\langle Conf(G), \sqsubseteq \rangle$ we have that:*

1. *if $C \in Conf(G)$ and $e \in C$, then $C[\![e]\!] \in Conf(G)$. Moreover $C[\![e]\!] \sqsubseteq C$;*
2. *if $C, C' \in Conf(G)$, $C \uparrow C'$ and $e \in C \cap C'$ then $C[\![e]\!] = C'[\![e]\!]$; in particular this holds for $C \sqsubseteq C'$;*
3. *if $e \in E$, $C_0, C_1 \in Hist(e)$ and $C_0 \uparrow C_1$ then $C_0 = C_1$.*

Proof. 1. Obviously, $C[\![e]\!] \in Conf(G)$. In fact, the requirements (1) and (2) of the definition of configuration are trivially satisfied, while (3) follows by recalling that $\nearrow \supseteq <$. Moreover $C[\![e]\!] \subseteq C$ and if $e' \in C[\![e]\!]$, $e'' \in C$ and $e'' \nearrow e'$, then $e'' \nearrow e' (\nearrow_C)^* e$, thus $e'' \in C[\![e]\!]$. Therefore $C[\![e]\!] \sqsubseteq C$.

2. By Lemma 4, since $C \uparrow C'$ and $e \in C$, an inductive reasoning ensures that if $e_0 \nearrow e_1 \nearrow \ldots \nearrow e_n \nearrow e$, with $e_i \in C \cup C'$, then each e_i is in C. Therefore $C[\![e]\!] = (C \cup C')[\![e]\!] = C'[\![e]\!]$.

3. Since $C_0 \uparrow C_1$ and $e \in C_0 \cap C_1$, by (2), we have

$$C_0 = C_0[\![e]\!] = C_1[\![e]\!] = C_1.$$

$\square$

We are now able to show that the complete primes of $Conf(G)$ are exactly the possible histories of events in G.

Lemma 7 (primes). *Let G be an* AES. *Then*

1. *for all configurations $C \in Conf(G)$*

$$C = \bigsqcup\{C' \in Hist(G) \mid C' \sqsubseteq C\} = \bigsqcup\{C[\![e]\!] \mid e \in C\}.$$

2. *$Pr(Conf(G)) = Hist(G)$ and $Pr(C) = \{C[\![e]\!] \mid e \in C\}$.*

Proof. 1. Let $C \in Conf(G)$ and let $C_0 = \bigsqcup\{C' \in Hist(G) \mid C' \sqsubseteq C\}$. Then clearly $C_0 \sqsubseteq C$. Moreover for all $e \in C$, by Lemma 6, the history $C[\![e]\!] \sqsubseteq C$ and thus $e \in C[\![e]\!] \subseteq C_0$. This gives the converse inclusion and allows us to conclude $C = C_0$.

2. Let $C[\![e]\!] \in Hist(e)$, for some $e \in E$, be a history and let $A \subseteq Conf(G)$ be a pairwise compatible set of configurations. If $C[\![e]\!] \sqsubseteq \bigsqcup A$, then $e \in \bigcup A$. Thus there exists $C_e \in A$ such that $e \in C_e$. Therefore:

$C[\![e]\!] = (\bigsqcup A)[\![e]\!]$	[by Lemma 6, since $C[\![e]\!] \sqsubseteq \bigsqcup A$,]
$= C_e[\![e]\!]$	[by Lemma 6, since $C_e \sqsubseteq \bigsqcup A$,]
$\sqsubseteq C_e$	[by Lemma 6].

Therefore $C[\![e]\!]$ is a complete prime in $Conf(G)$.

For the converse, let $C \in Pr(Conf(G))$. Then, by point (1),

$$C = \bigsqcup\{C' \in Hist(G) \mid C' \sqsubseteq C\}.$$

Being C a complete prime, there must exist $C' \in Hist(G)$, $C' \sqsubseteq C$ such that $C \sqsubseteq C'$ and thus $C = C' \in Hist(G)$. □

It is now immediate to prove that the configurations of an AES ordered by the extension relation form a finitary prime algebraic domain.

Theorem 3 (configurations form a domain). *Let G be an* AES*. Then $\langle Conf(G), \sqsubseteq\rangle$ is a (coherent finitary prime algebraic) domain.*

Proof. By Lemma 5, point 1, $Conf(G)$ is a coherent partial order. By Lemma 7, for any configuration $C \in Conf(G)$

$$Pr(C) = \{C[\![e]\!] \mid e \in C\}$$

and $C = \bigsqcup C[\![e]\!]$. Therefore $Conf(G)$ is prime algebraic.

Finally, $Conf(G)$ is finitary, as it immediately follows form the fact that compact elements in $Conf(G)$ are exactly the finite configurations. To see this, let $C \in Conf(G)$ be finite and let us consider a directed $A \subseteq Conf(G)$ such that $C \sqsubseteq \bigsqcup A$. Then we can choose, for all $e \in C$, $C_e \in A$ such that $e \in C_e$. Being A directed and C finite, the set $\{C_e \mid e \in C\}$ has an upper bound $C' \in A$. Then $C = \bigsqcup_{e \in C} C[\![e]\!] \sqsubseteq C'$ follows immediately from Lemma 6. Thus C is compact. For the converse, let $C \in Conf(G)$ be a compact element. Since each possible history is finite, $\{\bigcup_{e \in Z} C[\![e]\!] \mid Z \subseteq_{fin} C\}$ is a directed set of *finite* configurations, having C as least upper bound. Since C is compact, we conclude that there exists $Z \subseteq_{fin} C$ such that $C \sqsubseteq \bigcup_{e \in Z} C[\![e]\!]$. Thus $C = \bigcup_{e \in Z} C[\![e]\!]$ is finite. □

An example of AES with the corresponding domain can be found in Figure 8, (a) and (b), at the end of Section 7 (for the moment, the reader should disregard how the AES is obtained from the c-net N). In particular notice how asymmetric conflict influence the order on configurations, which is different from set-inclusion. For instance, $\{t_0, t_4\} \subseteq \{t_0, t_1', t_4\}$, but $\{t_0, t_4\} \not\sqsubseteq \{t_0, t_1', t_4\}$ since $t_1' \nearrow t_4$.

The next lemma gives a characterization of the immediate predecessors of a configuration. It states that, as one could expect, we pass from a configuration to one of its immediate successors by executing a single event.

Lemma 8 (immediate precedence). *Let G be an* AES *and let $C \sqsubseteq C'$ be configurations in $Conf(G)$. Then*

$$C \prec C' \quad \textit{iff} \quad |\, C' - C \,| = 1.$$

Proof. ($\Rightarrow$) Let $C \prec C'$ and let $e', e'' \in C' - C$. We have $C \sqsubset C \sqcup (C'[\![e']\!]) \sqsubseteq C'$ thus, by definition of immediate precedence, $C' = C \cup (C'[\![e']\!])$. In the same way $C' = C \cup C'[\![e'']\!]$. Thus $e' = e''$ (otherwise there would be $\nearrow_{C'}$-cycle $e' \nearrow e'' \nearrow e'$ in C').

($\Leftarrow$) Obvious. □

The following lemma leads to the definition of a functor from **AES** to **Dom**. First we prove that AES-morphisms preserve configurations and then we show that the natural function between the domains of configurations induced by an AES-morphism is a domain morphism.

Lemma 9 (AES**-morphisms preserve configurations).** *Let G_0, G_1 be two* AES*'s and let $f : G_0 \to G_1$ be an* AES*-morphism. Then for each $C_0 \in Conf(G_0)$ the morphism f is injective on C_0 and the f-image of C_0 is a configuration of G_1, i.e.,*

$$f^*(C_0) = \{f(e) \mid e \in C_0\} \in Conf(G_1).$$

Moreover $f^* : Conf(G_0) \to Conf(G_1)$ is a domain morphism.

Proof. Let $C_0 \in \mathit{Conf}(G_0)$ be a configuration. Since $\nearrow_{C_0}$ is well founded and thus $\neg(e \#^a e')$ for all $e, e' \in C_0$, the conditions of the definition of AES-morphism (Definition 4) imply that for all e, e' in C_0:

$$\lfloor f(e) \rfloor \subseteq f(\lfloor e \rfloor), \quad \text{if } f(e) \text{ is defined;}$$
$$f(e) = f(e') \quad \Rightarrow \quad e = e';$$
$$f(e) \nearrow_1 f(e') \quad \Rightarrow \quad e \nearrow_0 e'.$$

Therefore we immediately conclude that f is injective on C_0 (as expressed by the second condition) and $f^*(C_0)$ is a configuration in G_1.

Let us now prove that $f^* : \mathit{Conf}(G_0) \to \mathit{Conf}(G_1)$ is a domain morphism. Additivity and stability follow from Lemma 5. In particular for stability one should also observe that if C_0 and C_1 are compatible then f is injective on $C_1 \cup C_2$ and thus $f(C_1 \cap C_2) = f(C_1) \cap f(C_2)$. Finally, the fact that f^* preserves immediate precedence can be straightforwardly derived from Lemma 8. □

Theorem 3 and Lemma 9 suggest how to define a functor from the category **AES** of asymmetric event structures to the category **Dom** of domains. Instead, the functor going back from **Dom** to **AES** first transforms a domain into a PES via $\mathcal{P} : \mathbf{Dom} \to \mathbf{PES}$, introduced in Definition 11, and then embeds such PES into **AES** via $\mathcal{J} : \mathbf{PES} \to \mathbf{AES}$, defined in Proposition 1.

Definition 15 (from AES's to domains and backwards). *The functor $\mathcal{L}_a : \mathbf{AES} \to \mathbf{Dom}$ is defined as:*

- *for any* **AES**-*object G,*

$$\mathcal{L}_a(G) = \langle \mathit{Conf}(G), \sqsubseteq \rangle;$$

- *for any* **AES**-*morphism $f : G_0 \to G_1$,*

$$\mathcal{L}_a(f) = f^* : \mathcal{L}_a(G_0) \to \mathcal{L}_a(G_1).$$

The functor $\mathcal{P}_a : \mathbf{Dom} \to \mathbf{AES}$ is defined as $\mathcal{J} \circ \mathcal{P}$.

It is worth recalling that, concretely, given a domain $\langle D, \sqsubseteq \rangle$, the PES $\mathcal{P}(D)$ is defined as $\langle \mathrm{Pr}(D), \sqsubseteq, \# \rangle$, where $\#$ is the incompatibility relation (i.e., $p \# p'$ iff p and p' do not have a common upper bound). Then $\mathcal{P}_a(D) = \mathcal{J}(\mathcal{P}(D))$ is the corresponding AES, namely $\langle \mathrm{Pr}(D), \sqsubseteq, \sqsubseteq \cup \# \rangle$.

The functor $\mathcal{P}_a$ is left adjoint to $\mathcal{L}_a$ and they establish a coreflection between **AES** and **Dom**. The counit of the adjunction maps each history of an event e into the event e itself. The next technical lemma shows that the function defined in this way is indeed an AES-morphism.

Lemma 10. *Let G be an* AES. *Then $\epsilon_G : \mathcal{P}_a(\mathcal{L}_a(G)) \to G$ defined as:*

$$\epsilon_G(C) = e \qquad \text{if } C \in \mathit{Hist}(e),$$

is an AES-*morphism.*

Proof. Let us verify that ϵ_G satisfies the three conditions imposed on AES-morphisms: for all $C, C' \in \mathit{Hist}(G)$, with $C \in \mathit{Hist}(e)$, $C' \in \mathit{Hist}(e')$:

- $\lfloor \epsilon_G(C) \rfloor \subseteq \epsilon_G(\lfloor C \rfloor)$.
  We have:
  $$\begin{aligned} &\epsilon_G(\lfloor C \rfloor) = \\ &\quad = \epsilon_G(Pr(C)) \\ &\quad = \epsilon_G(\{C[\![e']\!] \mid e' \in C\}) && \text{[by Lemma 7]} \\ &\quad = C \\ &\quad \supseteq \lfloor e \rfloor && \text{[since } C \text{ is left-closed]} \\ &\quad = \lfloor \epsilon_G(C) \rfloor \end{aligned}$$
- $(\epsilon_G(C) = \epsilon_G(C')) \ \wedge \ C \neq C' \quad \Rightarrow \quad C \#^a C'$.
  Let $\epsilon_G(C) = e = e' = \epsilon_G(C')$ and $C \neq C'$. Since $C, C' \in \mathit{Hist}(e)$, by Lemma 6, we have $\neg(C \uparrow C')$ and thus $C \# C'$ in $\mathcal{P}(\mathcal{L}_a(G))$ and therefore, by definition of $\mathcal{J}$, $C \#^a C'$ in $\mathcal{P}_a(\mathcal{L}_a(G))$.

– $\epsilon_G(C) \nearrow \epsilon_G(C') \quad \Rightarrow \quad (C \nearrow C') \ \vee \ (C \#^a C')$.
Let $\epsilon_G(C) = e \nearrow e' = \epsilon_G(C')$. Being the relation $\nearrow$ irreflexive, surely $e \neq e'$ and thus $C \neq C'$. Now, if $e \notin C'$ then, by Lemma 4, surely $\neg(C \uparrow C')$, thus $C \# C'$ in $\mathcal{P}(\mathcal{L}_a(G))$ and therefore, by definition of $\mathcal{J}$, $C \#^a C'$ in $\mathcal{P}_a(\mathcal{L}_a(G))$. Otherwise, if $e \in C'$ we distinguish two cases:
  - $C = C[\![e]\!] = C'[\![e]\!]$.
    In this case, by Lemma 6, point 1, we have that $C \sqsubseteq C'$, and the relation is strict, since $C \neq C'$. Thus, by definition of $\mathcal{P}_a$, $C \nearrow C'$ in $\mathcal{P}_a(\mathcal{L}_a(G))$.
  - $C = C[\![e]\!] \neq C'[\![e]\!]$.
    In this case, by Lemma 6, point 2, we conclude that C and $C'[\![e]\!]$ are not compatible, and thus $\neg(C \uparrow C')$. Hence $C \# C'$ in $\mathcal{P}(\mathcal{L}_a(G))$ and therefore $C \#^a C'$ in $\mathcal{P}_a(\mathcal{L}_a(G))$.

□

The next technical lemma characterizes the behaviour of the functor $\mathcal{P}_a$ on morphisms having a domain of configurations as codomain.

Lemma 11. *Let G be an* AES*, D a domain and let $g : D \to \mathcal{L}_a(G)$ be a domain morphism. Then for all $p \in Pr(D)$, $\mid g(p) - \bigcup g(Pr(p) - \{p\}) \mid \leq 1$ and*

$$\mathcal{P}_a(g)(p) = \begin{cases} \bot & \text{if } g(p) - \bigcup g(Pr(p) - \{p\}) = \emptyset \\ g(p)[\![e]\!] & \text{if } g(p) - \bigcup g(Pr(p) - \{p\}) = \{e\} \end{cases}$$

Proof. Let $p \in Pr(D)$ and let us consider the corresponding prime interval

$$[\bigsqcup(Pr(p) - \{p\}), p],$$

then also

$$\left[g(\bigsqcup(Pr(p) - \{p\})), g(p)\right], \tag{1}$$

is a prime interval in $\mathcal{L}_a(G)$, and, by definition of the functor $\mathcal{E}_a$ (Definition 15)

$$\mathcal{P}_a(g)(p) = \begin{cases} \bot & \text{if } g(p) = g(\bigsqcup(Pr(p) - \{p\})) \\ C & \text{if } Pr(g(p)) - Pr(g(\bigsqcup(Pr(p) - \{p\}))) = \{C\} \end{cases}$$

Now, by additivity of g and Lemma 5.(1), $g(\bigsqcup(Pr(p) - \{p\})) = \bigsqcup g(Pr(p) - \{p\}) = \bigcup g(Pr(p) - \{p\})$, and, since (1) is a prime interval, by Lemma 8, $g(p) - \bigcup g(Pr(p) - \{p\})$ has at most one element. If $g(p) = \bigcup g(Pr(p) - \{p\})$ then $\mathcal{P}_a(g)(p) = \bot$. Otherwise, if $g(p) - \bigcup g(Pr(p) - \{p\}) = \{e\}$, then, by Lemma 7.(2), we have that $Pr(g(p)) - Pr(\bigcup g(Pr(p) - \{p\})) = \{g(p)[\![e]\!]\}$ and thus we conclude. □

Finally we can prove the main result of this section, namely, that $\mathcal{P}_a$ is left adjoint to $\mathcal{L}_a$ and they establish a coreflection between **AES** and **Dom**. Given an AES G, the component at G of the counit of the adjunction is $\epsilon_G : \mathcal{P}_a \circ \mathcal{L}_a(G) \to G$.

Theorem 4 (coreflection between AES and Dom). $\mathcal{P}_a \dashv \mathcal{L}_a$.

Proof. Let G be an AES and let $\epsilon_G : \mathcal{P}_a(\mathcal{L}_a(G)) \to G$ be the morphism defined as in Lemma 10. We have to show that given any domain D and AES-morphism $h : \mathcal{P}_a(D) \to G$, there is a unique domain morphism $g : D \to \mathcal{L}_a(G)$ such that the following diagram commutes:

$$\begin{tikzcd} \mathcal{P}_a(\mathcal{L}_a(G)) \arrow[r, "\epsilon_G"] & G \\ \mathcal{P}_a(D) \arrow[u, dotted, "\mathcal{P}_a(g)"] \arrow[ur, "h"'] & \end{tikzcd}$$

Existence
Let $g : D \to \mathcal{L}_a(G)$ be defined as:

$$g(d) = h^*(Pr(d)).$$

A straightforward checking shows that $Pr(d)$ is a configuration in $\mathcal{P}_a(D)$ and thus, by Lemma 9, h is injective on $Pr(d)$ and $h^*(Pr(d))$ is a configuration in G, i.e., an element of $\mathcal{L}_a(G)$. Moreover g is a domain morphism. In fact it is

- $\preceq$*-preserving.* Let $d, d' \in D$, with $d \prec d'$. Then $Pr(d') - Pr(d) = \{p\}$ and thus
  $$\begin{aligned} & g(d') - g(d) = \\ & \quad = h^*(Pr(d')) - h^*(Pr(d)) \\ & \quad \subseteq \{h(p)\} \end{aligned}$$
  Therefore $|g(d') - g(d)| \leq 1$ and, since it is easy to see that $g(d) \sqsubseteq g(d')$, by Lemma 8 we conclude $g(d) \preceq g(d')$.

- *Additive.* Let $X \subseteq D$ be a pairwise compatible set. Then:
  $$\begin{aligned} & g(\bigsqcup X) = \\ & \quad = h^*(Pr(\bigsqcup X)) \\ & \quad = h^*(\bigcup_{x \in X} Pr(x)) \qquad \text{[since } Pr(\bigsqcup X) = \bigcup_{x \in X} Pr(x)\text{]} \\ & \quad = \bigcup_{x \in X} h^*(Pr(x)) \\ & \quad = \bigsqcup_{x \in X} g(x) \end{aligned}$$
- *Stable.* Let $d, d' \in D$ with $d \uparrow d'$, then:
  $$\begin{aligned} & g(d \sqcap d') = \\ & \quad = h^*(Pr(d \sqcap d')) \\ & \quad = h^*(Pr(d) \cap Pr(d')) \qquad \text{[since } Pr(d \sqcap d') = Pr(d) \cap Pr(d') \text{ and } h \text{ injective on } Pr(d) \cup Pr(d')\text{]} \\ & \quad = h^*(Pr(d)) \cap h^*(Pr(d')) \\ & \quad = g(d) \sqcap g(d') \end{aligned}$$

The morphism g defined as above makes the diagram commute. In fact, let $p \in Pr(D)$ $(= \mathcal{P}_a(D))$ and let us use Lemma 11 to determine $\mathcal{P}_a(g)(p)$. We have:

$$\begin{aligned} & g(p) - \bigcup g(Pr(p) - \{p\}) = \\ & \quad = h^*(Pr(p)) - \bigcup\{h^*(Pr(p')) \mid p' \in Pr(D),\ p' \sqsubset p\} \\ & \quad = h^*(Pr(p)) - \{h(p'') \mid p'' \in Pr(D),\ p'' \sqsubset p\} \\ & \quad = h^*(Pr(p)) - h^*(Pr(p) - \{p\}) \\ & \quad = \{h(p)\} \qquad \text{[since } h \text{ injective on } Pr(p)\text{]} \end{aligned}$$

Therefore, if $h(p)$ is undefined then $\mathcal{P}_a(g)(p) = \bot$ and thus $\epsilon_G(\mathcal{P}_a(g)(p)) = \bot$. If $h(p) = e$ then $\mathcal{P}_a(g)(p) = g(p)[\![e]\!]$ and thus $\epsilon_G(\mathcal{P}_a(g)(p)) = e = h(p)$. Summing up we conclude

$$\epsilon_G \circ \mathcal{P}_a(g) = h.$$

Uniqueness
Let $g' : D \to \mathcal{L}_a(G)$ be another morphism such that

$$\epsilon_G \circ \mathcal{P}_a(g') = h.$$

By Lemma 11, for all $p \in Pr(D)$ we have:

$$\mathcal{P}_a(g')(p) = \begin{cases} \bot & \text{if } g'(p) - \bigcup g'(Pr(p) - \{p\}) = \emptyset \\ g'(p)[\![e]\!] & \text{if } g'(p) - \bigcup g'(Pr(p) - \{p\}) = \{e\} \end{cases}$$

Therefore

$$h(p) = \epsilon_G(\mathcal{P}_a(g')(p)) = \begin{cases} \bot & \text{if } g'(p) - \bigcup g'(Pr(p) - \{p\}) = \emptyset \\ e & \text{if } g'(p) - \bigcup g'(Pr(p) - \{p\}) = \{e\} \end{cases} \tag{2}$$

Let us show that $g'(p) = g(p)$ for all $p \in Pr(D)$, by induction on $k = |Pr(p)|$ (that is finite, since D is finitary).

$(\mathbf{k = 1})$ In this case $g'(p) - \bigcup g'(Pr(p) - \{p\}) = g'(p)$. Thus, by (2), if $h(p) = \bot$ then $g'(p) = \emptyset = g(p)$, otherwise, $g'(p) = \{h(p)\} = g(p)$.

$(\mathbf{k \to k + 1})$ First notice that being g' monotonic, for all $p' \in Pr(p)$ we have $g'(p') \sqsubseteq g'(p)$, thus

$$g'(p) = (g'(p) - (\bigcup g'(Pr(p) - \{p\}))) \cup (\bigcup g'(Pr(p) - \{p\})).$$

By inductive hypothesis, $\bigcup g'(Pr(p) - \{p\}) = \bigcup g(Pr(p) - \{p\})$, thus, reasoning as in the case $(k = 1)$ we conclude.

Being g and g' additive , since they coincide on the complete primes of D, which is prime algebraic, they coincide also on the whole domain D. □

4 Contextual nets

Contextual nets [28] extend classical Petri nets with the possibility of handling contexts: in a contextual net transitions can have not only preconditions and postconditions, but also *context* conditions. A transition can fire if enough tokens are present in its preconditions and context conditions. In the firing, preconditions are consumed, context conditions remains *unchanged* and new tokens are generated in the postconditions. This section introduces *(marked) contextual P/T nets* [34] (or *c-nets* for short), that following the lines suggested in [28] for C/E systems, add contexts to classical P/T nets.

To give the definition of c-net we need some notation for multisets. Let A be a set. A *multiset* of A is a function $M : A \to \mathbb{N}$. Such a multiset will be denoted sometimes as a formal sum $M = \sum_{a\in A} n_a \cdot a$, where $n_a = M(a)$. The set of multisets of A is denoted by μA. The usual operations and relations on multisets are used. For instance, multiset union is denoted by $+$ and defined as $(M + M')(a) = M(a) + M'(a)$; multiset difference $(M - M')$ is defined as $(M - M')(a) = M(a) - M'(a)$ if $M(a) \geq M'(a)$ and $(M - M')(a) = 0$ otherwise. We write $M \leq M'$ if $M(a) \leq M'(a)$ for all $a \in A$. If M is a multiset of A, we denote by $[\![M]\!]$ the multiset $\sum_{\{a\in A \mid M(a)>0\}} 1 \cdot a$, obtained by changing all non-zero coefficients of M to 1. Sometimes we will confuse the multisets $[\![M]\!]$ with the corresponding subsets $\{a \in A \mid M(a) > 0\}$ of A, and use on them the usual set operations and relations. A *multirelation* $f : A \to B$ is a multiset of $A \times B$. It induces in an obvious way a function $\mu f : \mu A \to \mu B$, defined as $\mu f(\sum_{a\in A} n_a \cdot a) = \sum_{b\in B} \sum_{a\in A}(n_a \cdot f(a, b)) \cdot b$. If the multirelation f satisfies $f(a, b) \leq 1$ for all $a \in A$ and $b \in B$ then we sometimes confuse it with the corresponding set-relation and write $f(a, b)$ for $f(a, b) = 1$. The composition of two multirelations $f : A \to B$ and $g : B \to C$ is the multirelation $g \circ f : A \to C$ defined as $(g \circ f)(a, c) = \sum_{b\in B} f(a, b) \cdot g(b, c)$.

We are now able to give the definition of contextual P/T net.

Definition 16 (c-net). *A* (marked) contextual Petri net (c-net) *is a tuple* $N = \langle S, T, F, C, m\rangle$, *where*

- S *is a set of* places*;*
- T *is a set of* transitions*;*
- $F = \langle F_{pre}, F_{post}\rangle$ *is a pair of multirelations, from* T *to* S.
- C *is a multirelation from* T *to* S, *called the* context relation*;*
- m *is a multiset of* S, *called the* initial marking.

We assume, without loss of generality, that $S \cap T = \emptyset$. *Moreover, we require that for each transition* $t \in T$, *there exists a place* $s \in S$ *such that* $F_{pre}(t, s) > 0$.[6]

In the following when considering a c-net N, we implicitly assume that $N = \langle S, T, F, C, m\rangle$. Moreover superscripts and subscripts on the nets names carry over the names of the involved sets, functions and relations. For instance $N_i = \langle S_i, T_i, F_i, C_i, m_i\rangle$.

Definition 17 (pre-set, post-set, and context). *Let* N *be a c-net. As usual, the functions from* μT *to* μS *induced by the multirelations* F_{pre} *and* F_{post} *are denoted by* ${}^\bullet(\ )$ *and* $(\ )^\bullet$, *respectively. If* $A \in \mu T$ *is a multiset of transitions,* ${}^\bullet A$ *is called its* pre-set, *while* $A^\bullet$ *is called its* post-set. *Moreover, by* $\underline{A}$ *we denote the* context *of* A, *defined as* $\underline{A} = \mu C(A)$.

An analogous notation is used to denote the functions from S *to* $\mathbf{2}^T$ *defined as, for* $s \in S$, ${}^\bullet s = \{t \in T \mid F_{post}(t, s) > 0\}$, $s^\bullet = \{t \in T \mid F_{pre}(t, s) > 0\}$, $\underline{s} = \{t \in T \mid C(t, s) > 0\}$.

For a multiset of transitions A to be enabled by a marking M, it is sufficient that M contains the pre-set of A and at least one *additional* token in each place in the context of A. This corresponds to the intuition that a token in a place can be used as context by many transitions at the same time and with multiplicity greater then one by the same transition.

[6] This is a weak version of the condition of *T-restrictness* that requires also $F_{post}(t, s) > 0$, for some $s \in S$.

Definition 18 (token game). *Let N be a c-net and let M be a* marking *of N, that is a multiset $M \in \mu S$. Given a finite multiset of transitions $A \in \mu T$, we say that A is* enabled *by M if ${}^\bullet A + [\![\underline{A}]\!] \leq M$. The* transition relation *between markings is defined as*

$$M\,[A\rangle\,M' \qquad \textit{iff} \qquad A \textit{ is enabled by } M \textit{ and } M' = M - {}^\bullet A + A^\bullet.$$

We call $M\,[A\rangle\,M'$ a step. *A* simple step *or a* firing *is a step involving just one transition, i.e., $M\,[t\rangle\,M'$.*

Other approaches (e.g. [19,35]) allow for the concurrent firing of transitions that use the same token as context and precondition.[7] We do not admit such steps, the idea being that two concurrent transitions should be allowed to fire also in any order.

Definition 19 (reachable marking). *Let N be a c-net. A marking M is called* reachable *if there exists a finite* step sequence

$$m\,[A_0\rangle\,M_1\,[A_1\rangle\,M_2 \ldots [A_n\rangle\,M$$

starting from the initial marking and leading to M.

A c-net morphism between two nets maps transitions and places of the first net into transitions and multisets of places of the second net, respectively, in such a way that the initial marking as well as the pre- and post-sets of transitions are "preserved". Contexts are preserved in a weak sense.

Definition 20 (c-net morphism). *Let N_0 and N_1 be c-nets. A* morphism *$h : N_0 \to N_1$ is a pair $h = \langle h_T, h_S\rangle$, where $h_T : T_0 \to T_1$ is a partial function and $h_S : S_0 \to S_1$ is a multirelation such that*

1. *$\mu h_S(m_0) = m_1$*
2. *for each $A \in \mu T$,*
   (a) $\mu h_S({}^\bullet A) = {}^\bullet \mu h_T(A)$;
   (b) $\mu h_S(A^\bullet) = \mu h_T(A)^\bullet$;
   (c) $[\![\underline{\mu h_T(A)}]\!] \leq \mu h_S(\underline{A}) \leq \underline{\mu h_T(A)}$.

We denote by **CN** *the category having c-nets as objects and c-net morphisms as arrows.*

Conditions (1), (2.a) and (2.b) are standard, but condition (2.c), regarding contexts, deserves some comments. It can be explained by recalling that, since in our model a single token can be used as context with multiplicity greater than one, the firing of a transition t can use as context any multiset X satisfying

$$[\![\underline{t}]\!] \leq X \leq \underline{t}.$$

Given any multiset of tokens that can be used as context in a firing of a transition, its image should be a set of tokens that can be used as context by the image of the transition. This can be formalized by requiring that $[\![\underline{\mu h_T(A)}]\!] \leq \mu h_S(X) \leq \underline{\mu h_T(A)}$ for any $X \in \mu S_0$ such that $[\![\underline{A}]\!] \leq X \leq \underline{A}$, which is equivalent to the above condition (2.c).

It is worth remarking that if h_T is undefined on a transition $t_0 \in T_0$, written $h_T(t_0) = \bot$, then, by definition of c-net morphism, the places in the pre-, post-set and context of t_0 are forced to be mapped to the empty set, i.e., $\mu h_S({}^\bullet t + t^\bullet + \underline{t}) = \emptyset$.

The basic result to prove (to check that the definition of morphism is "meaningful") is that the token game is preserved by c-net morphisms. As an immediate consequence morphisms preserve reachable markings.

Proposition 2 (morphisms preserve the token game). *Let N_0 and N_1 be c-nets, and let $h = \langle h_T, h_S\rangle : N_0 \to N_1$ be a morphism. Then for each $M, M' \in \mu S$ and $A \in \mu T$*

$$M\,[A\rangle\,M' \quad \Rightarrow \quad \mu h_S(M)\,[\mu h_T(A)\rangle\,\mu h_S(M').$$

Therefore c-net morphisms preserve reachable markings, i.e., if M_0 is a reachable marking in N_0 then $\mu h_S(M_0)$ is reachable in N_1.

[7] For instance, in [19] the formal condition for a multiset A of transition to be enabled by a marking M is ${}^\bullet A \leq M$ and $\underline{A} \leq M$.

Proof. First notice that $\mu h_T(A)$ is enabled by $\mu h_S(M)$. In fact, since A is enabled by M, we have $M \geq {}^\bullet A + [\![\underline{A}]\!]$. Thus

$$
\begin{aligned}
&\mu h_S(M) \\
&\quad \geq \mu h_S({}^\bullet A + [\![\underline{A}]\!]) \\
&\quad = \mu h_S({}^\bullet A) + \mu h_S([\![\underline{A}]\!]) \\
&\quad \geq \mu h_S({}^\bullet A) + [\![\mu h_S(\underline{A})]\!] \\
&\quad = {}^\bullet \mu h_T(A) + [\![\underline{\mu h_T(A)}]\!] \qquad \text{[by def. of c-net morphism]}
\end{aligned}
$$

Moreover $\mu h_S(M') = \mu h_S(M) - {}^\bullet \mu h_T(A) + \mu h_T(A)^\bullet$. In fact, $M' = M - {}^\bullet A + A^\bullet$, therefore we have:

$$
\begin{aligned}
&\mu h_S(M') \\
&\quad = \mu h_S(M) - \mu h_S({}^\bullet A) + \mu h_S(A^\bullet) \\
&\quad = \mu h_S(M) - {}^\bullet \mu h_T(A) + \mu h_T(A)^\bullet \qquad \text{[by def. of c-net morphism]}
\end{aligned}
$$

□

The seminal work by Winskel [39] presents a coreflection between prime event structures and a subclass of P/T nets, namely *safe* nets. In [26] it is shown that essentially the same constructions work for the larger category of "semi-weighted nets" as well (while the generalization to the whole category of P/T nets requires some original technical machinery and allows one to obtain a proper adjunction rather than a coreflection [25]). In the next sections we will relate by a coreflection (asymmetric and prime) event structures and "semi-weighted c-nets".

Definition 21 (semi-weighted c-nets). *A* semi-weighted *c-net is a c-net N such that the initial marking m is a set and F_{post} is a relation (i.e., $t^\bullet$ is a set for all $t \in T$). We denote by* **SW-CN** *the full subcategory of* **CN** *having semi-weighted c-nets as objects.*

A semi-weighted c-net is called safe *if also F_{pre} and C are relations (i.e., ${}^\bullet t$ and $\underline{t}$ are sets for all $t \in T$) and each reachable marking is a set. The full subcategory of* **SW-CN** *containing all safe c-nets is denoted by* **S-CN**.

Notice that the condition characterizing safe nets involves the dynamics of the net itself, while the one defining semi-weighted nets is "syntactical" in the sense that it can be checked statically, by looking only at structure of the net.

5 Occurrence c-nets

In the previous section we gave a description of the behaviour of a c-net in a dynamic way, by describing how the token game evolves. Occurrence c-nets are intended to represent, via the unfolding construction, the behaviour of general c-nets in a more static way, by expressing the events (firing of transitions) which can appear in a computation and the dependency relations between them. Occurrence c-nets will be defined as safe c-nets such that the dependency relations between transitions satisfy suitable acyclicity and well-foundedness requirements. While for traditional occurrence nets one has to take into account the causal dependency and the (symmetric) conflict relations, by the presence of contexts, we have to consider an asymmetric conflict (or weak dependency) relation as well. The conflict relation, as already seen in the more abstract setting of AES's, turns out to be a derived relation.

5.1 Dependency relations on transitions

Causal dependency is defined as for traditional safe nets, with an additional clause stating that transition t causes t' if it generates a token in a context place of t'.

Definition 22 (causal dependency). *Let N be a safe c-net. The* causal dependency relation *$<_N$ is the transitive closure of the relation $\prec$ defined by:*

1. *if $s \in {}^\bullet t$ then $s \prec t$;*
2. *if $s \in t^\bullet$ then $t \prec s$;*
3. *if $t^\bullet \cap \underline{t'} \neq \emptyset$ then $t \prec t'$.*

Given a place or transition $x \in S \cup T$, we denote with $\lfloor x \rfloor$ the set of causes *of x in T, defined as $\lfloor x \rfloor = \{t \in T \mid t \leq_N x\} \subseteq T$, where $\leq_N$ is the reflexive closure of $<_N$.*

Definition 23 (asymmetric conflict). *Let N be a safe c-net. The* strict asymmetric conflict relation *$\leadsto_N$ is defined as*

$$t \leadsto_N t' \qquad \textit{iff} \qquad \underline{t} \cap {}^\bullet t' \neq \emptyset \;\; \textit{or} \;\; (t \neq t' \;\wedge\; {}^\bullet t \cap {}^\bullet t' \neq \emptyset).$$

The asymmetric conflict relation *$\nearrow_N$ is the union of the strict asymmetric conflict and causal dependency relations:*

$$t \nearrow_N t' \qquad \textit{iff} \qquad t <_N t' \textit{ or } t \leadsto_N t'.$$

In our informal interpretation, $t \nearrow_N t'$ if t must precede t' in each computation in which both fire or, equivalently, t' prevents t to be fired:

$$t \nearrow t' \quad \stackrel{def}{\equiv} \quad \mathit{Fired}_C(t) \wedge \mathit{Fired}_C(t') \;\Rightarrow\; \mathit{prec}_C(t, t') \qquad (\dagger)$$

As noticed in the introduction, this is surely the case when the same place s appears as context for t and as precondition for t'. But ($\dagger$) is trivially true (with t and t' in interchangeable roles) when t and t' have a common precondition, since they never fire in the same computation. This is apparently a little tricky but corresponds to the clear intuition that a (usual) symmetric (direct) conflict leads to asymmetric conflict in both directions. Furthermore, since, as noticed for the general model of AES, ($\dagger$) is weaker than the condition that expresses causality, the condition ($\dagger$) is satisfied when t causes (in the usual sense) t'.[8] For technical reasons it is convenient to distinguish the first two cases from the last one.

In the sequel, when the net N is clear from the context, the subscripts in the relations $\leq_N$ and $\nearrow_N$ will be omitted.

The c-net N_4 in Figure 4 shows that, as expected, also in this setting the relation $\nearrow$ is not transitive. In fact we have $t_1 \nearrow t_3 \nearrow t_2 \nearrow t_1$, but, for instance, it is not true that $t_1 \nearrow t_2$.

N_4

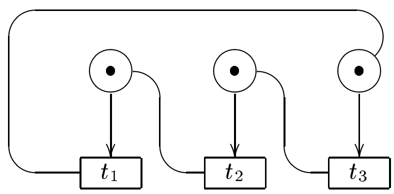


Fig. 4. An occurrence c-net with a cycle of asymmetric conflict.

An occurrence c-net is a safe c-net that exhibits an acyclic behaviour and such that each transition in it can be fired.

Definition 24 (occurrence c-nets). *An* occurrence c-net *is a safe c-net N such that*

1. *each place $s \in S$ is in the post-set of at most one transition, i.e., $|{}^\bullet s| \leq 1$;*
2. *the* causal relation *$<_N$ is irreflexive and its reflexive closure $\leq_N$ is a partial order such that $\lfloor t \rfloor$ is finite for any $t \in T$;*
3. *the* initial marking *m coincides with the set of minimal places with respect to $\leq_N$, i.e., $m = \{s \in S \mid {}^\bullet s = \emptyset\}$;*
4. *$(\nearrow_N)_{\lfloor t \rfloor}$ is acyclic[9] for all transitions $t \in T$.*

With **O-CN** *we denote the full subcategory of* **S-CN** *having occurrence c-nets as objects.*

[8] This is the origin of the weak causality interpretation of $\nearrow$.
[9] We can equivalently require $((\nearrow_N)_{\lfloor t \rfloor})^+$ to be irreflexive. In particular this implies $\nearrow_N$ irreflexive.

Conditions (1)-(3) are the same as for ordinary occurrence nets. Condition (4) corresponds to the requirement of irreflexivity for the conflict relation in ordinary occurrence nets. In fact, if a transition t has a $\nearrow_N$ cycle in its causes then it can never fire, since in an occurrence c-net N, the order in which transitions appear in a firing sequence must be compatible with the transitive closure of the (restriction to the transitions in the sequence of the) asymmetric conflict relation.

As anticipated the asymmetric conflict relation induces a symmetric conflict relation (on sets of transitions) defined in the following way:

Definition 25 (conflict). *Let N be a c-net. The* conflict *relation $\# \subseteq \mathbf{2}^T_{fin}$ associated to N is defined as:*

$$\frac{t_0 \nearrow t_1 \nearrow \ldots \nearrow t_n \nearrow t_0}{\#\{t_0, t_1, \ldots, t_n\}} \qquad\qquad \frac{\#(A \cup \{t\}) \quad t \leq t'}{\#(A \cup \{t'\})}$$

where A denotes a generic finite subset of T. As for AES*'s, we use the infix notation $t\#t'$ for $\#\{t, t'\}$.*

For instance, referring to Figure 4, we have $\#\{t_1, t_2, t_3\}$, but not $\#\{t_i, t_j\}$ for any $i, j \in \{1, 2, 3\}$. Notice that, by definition, the binary conflict relation $\#$ is symmetric. Moreover in an occurrence c-net $\#$ is irreflexive by the fourth condition of Definition 24.

Finally, observe that irreflexivity of the asymmetric conflict relation $\nearrow_N$ in an occurrence c-net N implies that the pre-set, the post-set and the context of any transition t in N are disjoint (any possible intersection would lead to $t \nearrow_N t$).

5.2 Concurrency and reachability

As for traditional occurrence nets, a set of places M is called concurrent if there is a reachable marking in which all the places of M contain a token. Differently from the classical case, for the presence of contexts some places that a transition needs to be fired (contexts) can be concurrent with the places it produces. However, the concurrency of a set of places can still be checked locally by looking only at the causes of such places and thus can be expressed via a "syntactical" condition. This subsection introduces such condition and then shows that it correctly formalizes the intuitive idea of concurrency.

Definition 26 (concurrency relation). *Let N be an occurrence c-net. A set of places $M \subseteq S$ is called* concurrent*, written $conc(M)$, if*

1. *$\forall s, s' \in M.\ \neg(s < s')$;*
2. *$\lfloor M \rfloor$ is finite, where $\lfloor M \rfloor = \bigcup\{\lfloor s \rfloor \mid s \in M\}$;*
3. *$\nearrow_{\lfloor M \rfloor}$ is acyclic (and thus well-founded, since $\lfloor M \rfloor$ is finite).*

In particular, for each transition t in an occurrence c-net, the set of places consisting of its pre-condition and context is concurrent.

Proposition 3. *For any transition t of an occurrence c-net, $conc({}^\bullet t + \underline{t})$.*

Proof. Since $\lfloor {}^\bullet t + \underline{t} \rfloor \cup \{t\} = \lfloor t \rfloor$, by definition of occurrence c-net, conditions (2) and (3) of concurrency are satisfied. As for the first condition, suppose that $s < s'$ for $s, s' \in {}^\bullet t + \underline{t}$. Then there is a transition t' such that $s \in {}^\bullet t'$ and $t' < s'$. Now, since $t' < s'$ and $s' \in {}^\bullet t + \underline{t}$ we have $t' < t$ and, since $s \in {}^\bullet t + \underline{t}$ and $s \in {}^\bullet t'$, we have also $t \nearrow t'$. Therefore $t' < t \nearrow t'$ is a $\nearrow$-cycle in $\lfloor t \rfloor$, contradicting the definition of occurrence c-net. Thus, also condition (1) is satisfied.

The next two lemmata show that given a concurrent set of places, we can interpret it as the result of a computation and perform a backward or forward step in such a computation, still obtaining a concurrent set.

Lemma 12 (backward steps preserve concurrency). *Let N be an occurrence c-net and let $M \subseteq S$ be a set of places. If $conc(M)$ and $t \in \lfloor M \rfloor$ is maximal with respect to $(\nearrow_{\lfloor M \rfloor})^+$ then*

1. *$\exists s_t \in S.\ s_t \in t^\bullet \cap M$;*

2. $conc(M - t^\bullet + {}^\bullet t)$.

Proof. 1. Since $t \in \lfloor M \rfloor$, there is $s_t \in M$ and $t' \in T$ such that $t \leq t'$ and $s_t \in t'^\bullet$. But recalling that $\nearrow$ implies $<$, by using maximality of t, we can conclude that $t = t'$.

2. Let $M' = M - t^\bullet + {}^\bullet t$. Clearly $\lfloor M' \rfloor = \lfloor M \rfloor - \{t\}$ and thus $\lfloor M' \rfloor$ is finite and $\nearrow_{\lfloor M' \rfloor}$ is acyclic. Moreover, we have to show there are no causal dependent (distinct) places in M'. Since $conc(M - t^\bullet)$, by hypothesis, and $conc({}^\bullet t)$, by Proposition 3, the only problematic case could be $s \in M - t^\bullet$ and $s' \in {}^\bullet t$. But
   - if $s < s'$ then, by transitivity of $<$, we have $s < s_t$;
   - if $s' < s$ then there is a transition t' such that $s' \in {}^\bullet t'$ and $t' \leq s$. Since $s' \in {}^\bullet t \cap {}^\bullet t'$, we have that $t \nearrow t' \nearrow t$ is a $\nearrow$-cycle in $\lfloor M \rfloor$.

   In both cases we reach a contradiction with the hypothesis $conc(M)$. □

Lemma 13 (forward steps preserve concurrency). *Let N be an occurrence c-net and let $M \subseteq S$ be a set of places. If $conc(M)$ and $M\,[t\rangle\, M'$ then $conc(M')$.*

Proof. The transition t is enabled by M, i.e., ${}^\bullet t + \underline{t} \subseteq M$ and thus $\neg(t \nearrow t')$ for all $t' \in \lfloor M \rfloor$. In fact let $t' \in \lfloor M \rfloor$, that is $t' < s'$ for some $s' \in M$. Clearly it can not be $t \rightsquigarrow t'$, otherwise, if $s \in {}^\bullet t' \cap ({}^\bullet t \cup \underline{t}) \subseteq M$ then $s < s'$, contradicting the hypothesis $conc(M)$. In the same way, if $t < t'$ then given any $s \in {}^\bullet t (\subseteq M)$, we would have $s < s'$.

Therefore, since $\lfloor M' \rfloor \subseteq \lfloor M \rfloor \cup \{t\}$ (the strict inclusion holds when $t^\bullet = \emptyset$) and, by hypothesis, $\nearrow_{\lfloor M \rfloor}$ is acyclic, we can conclude that $\nearrow_{\lfloor M' \rfloor}$ is acyclic. Moreover, being $\lfloor M \rfloor$ finite, also $\lfloor M' \rfloor$ is finite.

Finally, we have to show that there are no (distinct) causal dependent places in M'. Since $conc(M - {}^\bullet t)$ and $conc(t^\bullet)$ the only problematic case could be $s \in M - {}^\bullet t$ and $s' \in t^\bullet$. But

- if $s < s'$ then $s < s''$ for some $s'' \in {}^\bullet t \cup \underline{t}$;
- if $s' < s$ then, for $s'' \in {}^\bullet t$, by transitivity of $<$, $s'' < s$.

In both cases we reach a contradiction with the hypothesis $conc(M)$. □

It is now quite easy to conclude that, as mentioned before, the concurrent sets of places of a c-net indeed coincide with the (subsets of) reachable markings.

Proposition 4 (concurrency and reachability). *Let N be an occurrence c-net and let $M \subseteq S$ be a set of places. Then*

$$conc(M) \qquad \textit{iff} \qquad M \subseteq M' \textit{ for some reachable marking } M'.$$

Proof. ($\Rightarrow$) By definition of the concurrency relation, $\lfloor M \rfloor$ is finite. Moreover $\nearrow_{\lfloor M \rfloor}$ is acyclic and therefore there is an enumeration $t^{(1)}, \ldots, t^{(k)}$ of the transitions in $\lfloor M \rfloor$ compatible with $(\nearrow_{\lfloor M \rfloor})^+$. Let us show by induction on $k = |\lfloor M \rfloor|$ that

$$m = M^{(0)}\,[t^{(1)}\rangle\, M^{(1)}\,[t^{(2)}\rangle\, M^{(2)} \ldots [t^{(k)}\rangle\, M^{(k)} \supseteq M.$$

($\mathbf{k = 0}$) In this case simply $m \supseteq M$ and obviously $conc(m)$.
($\mathbf{k > 0}$) By construction, $t^{(k)}$ is maximal in $\lfloor M \rfloor$ with respect to $(\nearrow_{\lfloor M \rfloor})^+$. Thus, by Lemma 12, if we define $M'' = M - t^{(k)\bullet} + {}^\bullet t^{(k)}$, we have $conc(M'')$ and $\lfloor M'' \rfloor = \{t^{(1)}, \ldots, t^{(k-1)}\}$. Therefore, by inductive hypothesis, there is a firing sequence

$$m\,[t^{(1)}\rangle\, M^{(1)} \ldots [t^{(k-1)}\rangle\, M^{(k-1)} \supseteq M''. \tag{3}$$

Now, by construction, ${}^\bullet t^{(k)} \subseteq M''$. Moreover also $\underline{t^{(k)}} \subseteq M''$. In fact, if $s \in \underline{t^{(k)}}$ then $s \in m$ or $s \in t^{(h)\bullet}$ for some $h < k$. Thus a token in s is generated in the firing sequence (3), and no transition $t^{(l)}$ can consume this token, otherwise $t^{(k)} \nearrow t^{(l)}$, contradicting the maximality of $t^{(k)}$. Finally, by definition of occurrence c-net, ${}^\bullet t^{(k)} \cap \underline{t^{(k)}} = \emptyset$, being $\nearrow$ irreflexive. Therefore $t^{(k)}$ is enabled in M'' so that we can extend the firing sequence (3) to

$$m\,[t^{(1)}\rangle\, M^{(1)} \ldots [t^{(k-1)}\rangle\, M^{(k-1)}\, [t^{(k)}\rangle\, M^{(k)},$$

where $M^{(k)} = M^{(k-1)} - {}^\bullet t^{(k)} + t^{(k)\bullet} \supseteq M'' - {}^\bullet t^{(k)} + t^{(k)\bullet} = M$.

($\Leftarrow$) Let us suppose that there exists a firing sequence

$$m\,[t^{(1)}\rangle\, M^{(1)}\, [t^{(2)}\rangle\, M^{(2)} \ldots [t^{(k)}\rangle\, M^{(k)} \supseteq M.$$

and let us prove that $conc(M^{(k)})$ (and thus $conc(M)$).

If $(k = 0)$, then $M \subseteq m$ and clearly $conc(m)$. If $k > 0$ then an inductive reasoning that uses Lemma 13 allows one to conclude. □

As an immediate corollary we obtain that each transition of an occurrence c-net is firable in some computation of the net.

Corollary 1. *For any transition t of an occurrence c-net N there is a reachable marking M of N which enables t.*

Proof. By Proposition 3, $conc({}^\bullet t + \underline{t})$ and thus, by Proposition 4, we can find a reachable marking M of N, such that $M \supseteq {}^\bullet t + \underline{t}$, enabling t. □

5.3 Morphisms on occurrence c-nets

This subsection states some properties of c-net morphisms between occurrence c-nets that will be useful in the sequel. We start with a characterization of such morphisms.

Lemma 14 (occurrence c-nets morphisms). *Let N_0 and N_1 be occurrence c-nets and let $h = \langle h_T, h_S\rangle : N_0 \to N_1$ be a morphism. Then h_S is a* relation *and*

- $\forall s_1 \in m_1.\ \exists! s_0 \in m_0.\ h_S(s_0, s_1)$;
- *for each $t_0 \in T_0$, if $h_T(t_0) = t_1$ then*
  - $\forall s_1 \in {}^\bullet t_1.\ \exists! s_0 \in {}^\bullet t_0.\ h_S(s_0, s_1)$;
  - $\forall s_1 \in \underline{t_1}.\ \exists! s_0 \in \underline{t_0}.\ h_S(s_0, s_1)$;
  - $\forall s_1 \in t_1{}^\bullet.\ \exists! s_0 \in t_0{}^\bullet.\ h_S(s_0, s_1)$;

Moreover given any $s_0 \in S_0$, $s_1 \in S_1$, $t_1 \in T_1$:

- $s_1 \in m_1 \ \wedge\ h_S(s_0, s_1) \quad \Rightarrow \quad s_0 \in m_0$;
- $s_1 \in t_1{}^\bullet \wedge\ h_S(s_0, s_1) \quad \Rightarrow \quad \exists! t_0 \in T_0.\ (s_0 \in t_0{}^\bullet \ \wedge\ h_T(t_0) = t_1)$.

Proof. The result is easily proved by using the structural properties of occurrence c-nets. We treat just the first point. Let $s_1 \in m_1$. Since it must be $\mu h_S(m_0) = m_1$, there exists $s_0 \in S_0$ such that $h_S(s_0, s_1)$. Such s_0 must be unique, since otherwise the initial marking of N_1 should be a proper multiset, rather than a set, contradicting the definition of occurrence c-net. □

As an easy consequence of the results in the previous subsection, c-net morphisms preserve the concurrency relation.

Corollary 2 (morphisms preserve concurrency). *Let N_0 and N_1 be occurrence c-nets and let $h : N_0 \to N_1$ be a morphism. Given $M_0 \subseteq S_0$, if $conc(M_0)$ then $\mu h_S(M_0)$ is a set and $conc(\mu h_S(M_0))$.*

Proof. Let $M_0 \subseteq S_0$, with $conc(M_0)$. Then, by Proposition 4, there exists a firing sequence in N_0:

$$m_0\,[t^{(1)}\rangle\, M^{(1)} \ldots [t^{(n)}\rangle\, M^{(n)} \supseteq M_0.$$

By Proposition 2, morphisms preserve the token game and thus

$$m_1 = \mu h_S(m_0)\,[h_T(t^{(1)})\rangle\, \mu h_S(M^{(1)}) \ldots [h_T(t^{(n)})\rangle\, \mu h_S(M^{(n)}) \supseteq \mu h_S(M_0).$$

is a firing sequence in N_1. Hence $\mu h_S(M_0)$ is a set and, by Proposition 4, $conc(\mu h_S(M_0))$. □

Notice that the corollary implicitly states that morphisms are "injective" on concurrent sets, in the sense that if $conc(M)$ and $s \neq s'$ are in M, then $\mu h_S(s)$, $\mu h_S(s')$ are sets and $\mu h_S(s) \cap \mu h_S(s') = \emptyset$ (otherwise $\mu_S(M)$ would be a proper multiset).

The next lemmata show that, more generally, morphisms preserve the "amount of concurrency". More precisely first we prove that morphisms reflect $\prec$-chains.[10] Then, after proving some other technical results, we can conclude that morphisms reflect causal dependency and conflicts, while asymmetric conflict is reflected or becomes conflict.

Lemma 15 (morphisms reflect $\prec$-chains). *Let N_0 and N_1 be occurrence c-nets, let $h : N_0 \to N_1$ be a morphism and let $t_1^{(0)} \prec t_1^{(1)} \prec \ldots \prec t_1^{(n)}$ be a chain of transitions in T_1 such that $t_1^{(n)} = h_T(t_0^{(n)})$. Then there exists a chain $t_0^{(0)} \prec t_0^{(1)} \prec \ldots \prec t_0^{(n)}$ in T_0 such that $t_1^{(i)} = h_T(t_0^{(i)})$ for all $i \in \{0, \ldots, n\}$.*

Proof. We proceed by induction on n:

$(\mathbf{n = 0})$ Obvious.

$(\mathbf{n > 0})$ Let $t_1^{(0)} \prec t_1^{(1)} \prec \ldots \prec t_1^{(n)} = h_T(t_0^{(n)})$. By inductive hypothesis (applied to the final part of the chain) there is a chain $t_0^{(1)} \prec \ldots \prec t_0^{(n)}$ such that $t_1^{(i)} = h_T(t_0^{(i)})$ for $i \in \{1, \ldots, n\}$.

Moreover, since $t_1^{(0)} \prec t_1^{(1)} = h_T(t_0^{(1)})$, two cases arise:

- $t_1^{(0)\bullet} \cap \underline{t_1^{(1)}} \neq \emptyset$.
  Let $s_1 \in t_1^{(0)\bullet} \cap \underline{t_1^{(1)}}$. Since $t_1^{(1)} = h_T(t_0^{(1)})$ there is $s_0 \in \underline{t_0^{(1)}}$ such that $f_S(s_0, s_1)$. Moreover, by Lemma 14, from $s_1 \in t_1^{(0)\bullet}$ we have that that $\exists! t_0^{(0)} \in T_0$ such that $h_T(t_0^{(0)}) = t_1^{(0)}$ and $s_0 \in t_0^{(0)\bullet}$. Therefore $t_0^{(0)} \prec t_0^{(1)}$ is the transition that completes the chain.

- $t_1^{(0)\bullet} \cap {}^\bullet t_1^{(1)} \neq \emptyset$.
  Analogous to the previous case.

$\square$

Lemma 16. *Let N_0 and N_1 be occurrence c-nets and let $h : N_0 \to N_1$ be a morphism. Then, for all $t_0, t_0' \in T_0$*

1. *$(h_T(t_0) = h_T(t_0')) \ \wedge \ (t_0 \neq t_0') \quad \Rightarrow \quad t_0 \#_0 t_0'$;*
2. *$h_T(t_0) \leq_1 h_T(t_0') \quad \Rightarrow \quad \exists! t_0'' \in T_0.\ (t_0'' \leq_0 t_0' \ \wedge \ h_T(t_0'') = h_T(t_0))$;*
3. *(a) $h_T(t_0) \leadsto_1 h_T(t_0') \quad \Rightarrow \quad (t_0 \leadsto_0 t_0') \ \vee \ (t_0 \#_0 t_0')$;*
   (b) $h_T(t_0) <_1 h_T(t_0') \quad \Rightarrow \quad (t_0 <_0 t_0') \ \vee \ (t_0 \#_0 t_0')$;

Proof. 1. Let $h_T(t_0) = h_T(t_0')$ and $t_0 \neq t_0'$. Consider a chain of transitions $t_1^{(0)} \prec \ldots \prec t_1^{(k)} = h_T(t_0)$ such that ${}^\bullet t_1^{(0)} \subseteq m_1$ and $t_1^{(i)\bullet} \cap {}^\bullet t_1^{(i+1)} \neq \emptyset$ for all $i \in \{0, \ldots, k-1\}$ (the existence of such a finite chain is an immediate consequence of the definition of occurrence c-net). By Lemma 15, we can find in T_0 two $\prec$-chains of transitions,

$$t_0^{(0)} \prec \ldots \prec t_0^{(k)} \qquad \text{and} \qquad t_0'^{(0)} \prec \ldots \prec t_0'^{(k)},$$

such that, $h_T(t_0^{(i)}) = h_T(t_0'^{(i)}) = t_1^{(i)}$, for all $i \in \{1, \ldots, k\}$ and $t_0 = t_0^{(k)}$, $t_0' = t_0'^{(k)}$.
Let j be the least index such that $t_0^{(j)} \neq t_0'^{(j)}$. If $j = 0$ (and thus ${}^\bullet t_1^{(j)} \subseteq m_1$) consider a generic $s_1 \in {}^\bullet t_1^{(0)}$. By definition of morphism, there are $s_0 \in {}^\bullet t_0^{(0)}$ and $s_0' \in {}^\bullet t_0'^{(0)}$ such that $h_S(s_0, s_1)$ and $h_S(s_0', s_1)$. By Lemma 14, since $s_1 \in m_1$, also s_0 and s_0' are in the initial marking and thus $s_0 = s_0'$. Hence $t_0^{(0)} \nearrow_0 t_0'^{(0)} \nearrow_0 t_0^{(0)}$, thus $t_0^{(0)} \#_0^m t_0'^{(0)}$ and therefore, by definition of $\#$, $t_0 \#_0 t_0'$. If $j > 0$, then considering $s_1 \in t_1^{(j-1)\bullet} \cap {}^\bullet t_1^{(j)}$, the same reasoning applies.

[10] The symbol $\prec$ is used here to denote immediate causal dependency between transitions. Formally $t \prec t'$ if $t < t'$ and there does not exists t'' such that $t < t'' < t'$.

2. Existence easily follows from Lemma 15. As for uniqueness, let $t_0''' \leq_0 t_0'$ and $h_T(t_0''') = h_T(t_0)$. If $t_0''' \neq t_0''$ then, by point (1.a) $t_0''' \#_0 t_0''$ and therefore $t_0' \#_0 t_0'$, contradicting the definition of occurrence c-net.
3. (a) Let $h_T(t_0) \rightsquigarrow_1 h_T(t_0')$. Then there is a place $s_1 \in (\underline{h_T(t_0)} \cup {}^\bullet h_T(t_0)) \cap {}^\bullet h_T(t_0')$. Thus there are $s_0 \in (\underline{t_0} \cup {}^\bullet t_0)$ such that $h_S(s_0, s_1)$ and $s_0' \in {}^\bullet t_0'$ such that $h_S(s_0', s_1)$. If s_1 is in the initial marking then $s_0 = s_0'$ and thus $t_0 \rightsquigarrow_1 t_0'$. Otherwise s_0 and s_0' are in the post-sets of two transitions $t_0^{(0)}$ and $t_0'^{(0)}$, which are mapped to the same transition in N_1 (the transition which has s_1 in its post-set). By point (1), $t_0^{(0)}$ and $t_0'^{(0)}$ are identical or in conflict: in the first case $s_0 = s_0'$ and thus $t_0 \rightsquigarrow_0 t_0'$, while in the second case $t_0 \#_0 t_0'$.
   (b) Let $h_T(t_0) <_1 h_T(t_0')$. By Lemma 15, there exists $t_0'' \in T_0$ such that $t_0'' <_0 t_0'$ and $h_T(t_0'') = h_T(t_0)$. It follows from point (1) that either $t_0'' = t_0$ and thus $t_0 <_0 t_0'$, or $t_0'' \#_0 t_0$ and thus $t_0 \#_0 t_0'$.

□

Corollary 3. *Let N_0 and N_1 be occurrence c-nets and let $h : N_0 \to N_1$ be a morphism. Then, for all $t_0, t_0' \in T_0$*

1. $\lfloor h_T(t_0) \rfloor \subseteq h_T(\lfloor t_0 \rfloor)$;
2. $(h_T(t_0) = h_T(t_0')) \ \wedge \ (t_0 \neq t_0') \quad \Rightarrow \quad t_0 \#_0 t_0'$;
3. $h_T(t_0) \nearrow_1 h_T(t_0') \quad \Rightarrow \quad (t_0 \nearrow_0 t_0') \ \vee \ (t_0 \#_0 t_0')$;
4. $\# h_T(A) \quad \Rightarrow \quad \# A'$, *for some* $A' \subseteq A$.

6 Unfolding: from semi-weighted to occurrence contextual nets

This section shows how, given a semi-weighted c-net N, an *unfolding* construction allows us to obtain an occurrence c-net $\mathcal{U}_a(N)$ that describes the behaviour of N. As for traditional nets, each transition in $\mathcal{U}_a(N)$ represents an instance of a precise firing of a transition in N, and places in $\mathcal{U}_a(N)$ represent occurrences of tokens in the places of N. Each item (place or transition) of the unfolding is mapped to the corresponding item of the original net by a c-net morphism $f_N : \mathcal{U}_a(N) \to N$, called the folding morphism. The unfolding operation can be extended to a functor $\mathcal{U}_a$: **SW-CN** $\to$ **O-CN** that is right adjoint to the inclusion functor $\mathcal{I}_O$: **O-CN** $\to$ **SW-CN** and thus establishes a coreflection between **SW-CN** and **O-CN**.

We first introduce some technical notions. We say that a c-net N_0 is a *subnet* of N_1, written $N_0 \trianglelefteq N_1$, if $S_0 \subseteq S_1$, $T_0 \subseteq T_1$ and the inclusion $\langle i_T, i_S \rangle$ (with $i_T(t) = t$, for all $t \in T_0$, and $i_S(s,s) = 1$, for all $s \in S_0$) is a c-net morphism. In words, $N_0 \trianglelefteq N_1$ if N_0 coincides with an initial segment of N_1. In the following it will be useful to consider the subnets of an occurrence c-net obtained by truncating the original net at a given "causal depth", where the notion of depth is defined in the natural way.

Definition 27 (depth). *Let N be an occurrence c-net. The function depth : $S \cup T \to \mathbb{N}$ is defined inductively as follows:*

$$\begin{array}{ll} depth(s) = 0 & \text{for } s \in m; \\ depth(t) = \max\{depth(s) \mid s \in {}^\bullet t \cup \underline{t}\} + 1 & \text{for } t \in T; \\ depth(s) = depth(t) & \text{for } s \in t^\bullet. \end{array}$$

It is not difficult to prove that *depth* is a well-defined total function, since infinite descending chains of causality are disallowed in occurrence c-nets. Moreover, given an occurrence c-net N, the net containing only the items of *depth* less or equal to k, denoted by $N^{[k]}$, is a well-defined occurrence c-net and it is a subnet of N. The following simple result holds:

Proposition 5. *An occurrence c-net N is the (componentwise) union of its subnets $N^{[k]}$, of depth k.*

The unfolding of a semi-weighted c-net N can be constructed inductively by starting from the initial marking of N, and then by adding, at each step, an instance of each transition of N enabled by (the image of) a concurrent subset of places in the partial unfolding currently generated. For technical reasons we prefer to give an axiomatic definition.

Definition 28 (unfolding). *Let $N = \langle S, T, F, C, m \rangle$ be a semi-weighted c-net. The unfolding $\mathcal{U}_a(N) = \langle S', T', F', C', m' \rangle$ of the net N and the* folding morphism $f_N = \langle f_T, f_S \rangle : \mathcal{U}_a(N) \to N$ *are the unique occurrence c-net and c-net morphism satisfying the following equations.*

$$\begin{array}{ll}
m' = \{\langle \emptyset, s \rangle \mid s \in m\} & \\
S' = m' \cup \{\langle t', s \rangle \mid t' = \langle M_p, M_c, t \rangle \in T' \ \wedge \ s \in t^{\bullet}\} & \\
T' = \{\langle M_p, M_c, t \rangle \mid M_p, M_c \subseteq S' \ \wedge \ M_p \cap M_c = \emptyset \ \wedge \ conc(M_p \cup M_c) \ \wedge & \\
\qquad t \in T \ \wedge \ \mu f_S(M_p) = {}^{\bullet}t \ \wedge \ [\![\underline{t}]\!] \leq \mu f_S(M_c) \leq \underline{t}\} &
\end{array}$$

$$\begin{array}{lll}
F'_{pre}(t', s') & \text{iff} & t' = \langle M_p, M_c, t \rangle \ \wedge \ s' \in M_p \quad (t \in T) \\
C'(t', s') & \text{iff} & t' = \langle M_p, M_c, t \rangle \ \wedge \ s' \in M_c \quad (t \in T) \\
F'_{post}(t', s') & \text{iff} & s' = \langle t', s \rangle \quad (s \in S)
\end{array}$$

$$\begin{array}{lll}
f_T(t') = t & \text{iff} & t' = \langle M_p, M_c, t \rangle \\
f_S(s', s) & \text{iff} & s' = \langle x, s \rangle \quad (x \in T' \cup \{\emptyset\})
\end{array}$$

The existence of the unfolding can be proved by explicitly giving its inductive definition. Uniqueness follows from the fact that each item in a occurrence c-net has a finite depth.

Places and transitions in the unfolding of a c-net represent respectively tokens and firing of transitions in the original net. Each place in the unfolding is a pair recording the "history" of the token and the corresponding place in the original net. Each transition is a triple recording the precondition and context used in the firing, and the corresponding transition in the original net. A new place with empty history $\langle \emptyset, s \rangle$ is generated for each place s in the initial marking of N. Moreover a new transition $t' = \langle M_p, M_c, t \rangle$ is inserted in the unfolding whenever we can find a concurrent set of places (precondition M_p and context M_c) that corresponds, in the original net, to a marking that enables t. For each place s in the post-set of such t, a new place $\langle t', s \rangle$ is generated, belonging to the post-set of t'. The folding morphism f maps each place (transition) of the unfolding to the corresponding place (transition) in the original net. Figure 5 shows a c-net N and the the initial part of its unfolding (formally, it is the subnet of the unfolding of depth 3, namely $\mathcal{U}_a(N)^{[3]}$). The folding morphism is represented by labelling the items of the unfolding with the names of the corresponding items of N.

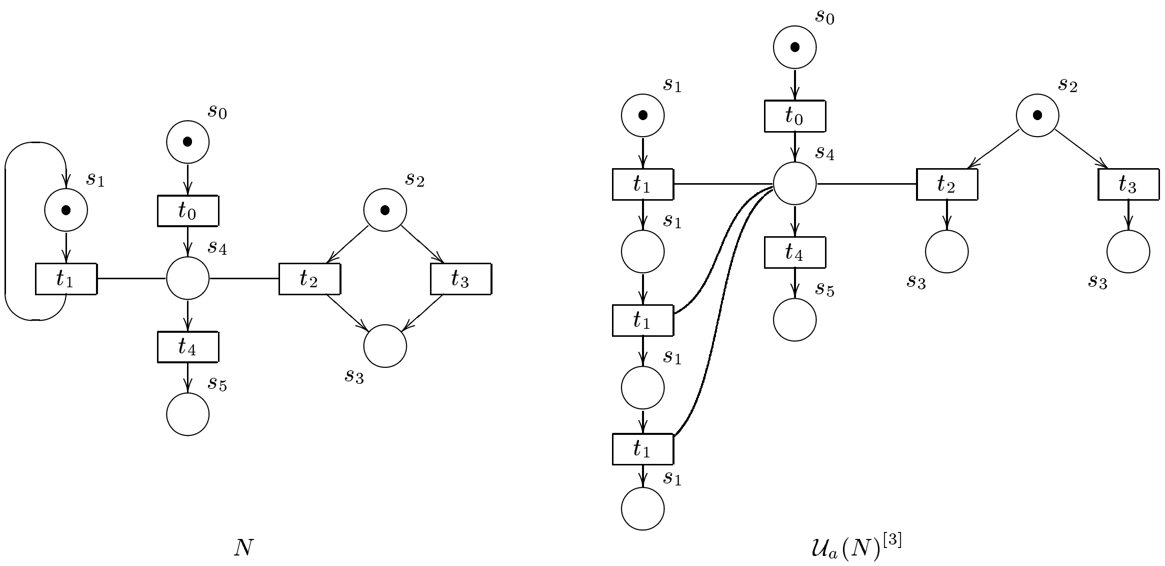


Fig. 5. A c-net and (part of) its unfolding.

Occurrence c-nets are particular semi-weighted c-nets, thus we can consider the inclusion functor $\mathcal{I}_O : \textbf{O-CN} \to \textbf{SW-CN}$ that acts as identity on objects and morphisms. We show now that the unfolding of a c-net $\mathcal{U}_a(N)$ and the folding morphism f_N are cofree over N. Therefore $\mathcal{U}_a$ extends to a functor that is right adjoint of $\mathcal{I}_O$ and thus establishes a coreflection between **SW-CN** and **O-CN**.

Theorem 5 (coreflection between SW-CN and O-CN). $\mathcal{I}_O \dashv \mathcal{U}_a$

Proof. Let N be a semi-weighted c-net, let $\mathcal{U}_a(N) = \langle S', T', F', C', m' \rangle$ be its unfolding and let $f_N : \mathcal{U}_a(N) \to N$ be the folding morphism as in Definition 28. We have to show that for any occurrence c-net N_1 and for any morphism $g : N_1 \to N$ there exists a unique morphism $h : N_1 \to \mathcal{U}_a(N)$ such that the following diagram commutes:

$$\begin{tikzcd} \mathcal{U}_a(N) \arrow[r, "f_N"] & N \\ N_1 \arrow[u, dotted, "h"] \arrow[ur, "g"'] & \end{tikzcd}$$

Existence
We define a sequence of morphisms $h^{[k]} : {N_1}^{[k]} \to \mathcal{U}_a(N)$ such that, for any k,

$$h^{[k]} \subseteq h^{[k+1]} \qquad \text{and} \qquad f_N \circ h^{[k]} = g_{|{N_1}^{[k]}},$$

then the morphism h we are looking for will be $h = \bigcup_k h^{[k]}$. We give an inductive definition:

$(\mathbf{k = 0})$ The c-net ${N_1}^{[0]}$ consists only of the initial marking of N_1 with no transitions, i.e., ${N_1}^{[0]} = \langle m_1, \emptyset, \emptyset, \emptyset, m_1 \rangle$. Therefore $h^{[0]}$ has to be defined:

$${h_T}^{[0]} = \emptyset,$$
$${h_S}^{[0]}(s_1, \langle \emptyset, s \rangle) \equiv g_S(s_1, s) \qquad \text{for all } s_1 \in {S_1}^{[0]} = m_1 \text{ and } s \in S.$$

$(\mathbf{k \to k+1})$ The morphism $h^{[k+1]}$ extends $h^{[k]}$ on items with depth equal to $k{+}1$ as follows. Let $t_1 \in T^{[k+1]}$ with $\mathit{depth}(t_1) = k+1$. By definition of depth, $\mathit{depth}(s) \le k$ for all $s \in {}^\bullet t_1 \cup \underline{t_1}$ and thus $h^{[k]}$ is defined on the pre-set and on the context of t_1. We must define h_T on t_1 and h_S on its post-set. Two cases arise:

- If $g_T(t_1) = \bot$ then necessarily
  $${h_T}^{[k+1]}(t_1) = \bot$$
  $${h_S}^{[k+1]}(s_1, s') = 0 \qquad \text{for all } s_1 \in {t_1}^\bullet \text{ and } s' \in S'.$$

- If $g_T(t_1) = t$ then consider the sets

  $$M_p = \mu {h_S}^{[k]}({}^\bullet t_1) \qquad\qquad M_c = \mu {h_S}^{[k]}(\underline{t_1}).$$

  Since N_1 is an occurrence c-net, ${}^\bullet t_1 \cap \underline{t_1} = \emptyset$ and, by Proposition 3, $\mathit{conc}({}^\bullet t_1 \cup \underline{t_1})$. Hence, by Corollary 2,

  $$M_p \cap M_c = \emptyset \qquad \text{and} \qquad \mathit{conc}(M_p \cup M_c).$$

  Moreover, by construction, $f_N \circ h^{[k]} = g_{|{N_1}^{[k]}}$, and therefore
  $$\begin{aligned} &\mu f_S(M_p) \\ &\quad = \mu f_S(\mu {h_S}^{[k]}({}^\bullet t_1)) \\ &\quad = \mu g_S({}^\bullet t_1) \\ &\quad = {}^\bullet t && \text{[by def. of morphism]} \end{aligned}$$
  and in the same way $[\![\underline{t}]\!] \le \mu f_S(M_c) \le \underline{t}$. Thus, by definition of unfolding, there exists a transition $t' = \langle M_p, M_c, t \rangle$ in T'.
  It is clear that, to obtain a well defined morphism that makes the diagram commute, we have to define

  $${h_T}^{[k+1]}(t_1) = t'$$

  and, since $\mu g_S({t_1}^\bullet) = t^\bullet$, for all $s_1 \in {t_1}^\bullet$ and $s \in t^\bullet$

$$h_S{}^{[k+1]}(s_1, \langle t', s\rangle) = g_S(s_1, s).$$

A routine checking allows to prove that, for each k, $h^{[k]}$ is a well-defined morphism and $f_N \circ h^{[k]} = g_{|N_1{}^{[k]}}$.

Uniqueness
The morphism h is clearly unique since at each step we were forced to define it as we did to ensure commutativity. Formally, let $h' : N_1 \to \mathcal{U}_a(N)$ a morphism such that the diagram commutes, i.e., $f_N \circ h' = g$. Then, we show, that for all k

$$h'_{|N_1{}^{[k]}} = h_{|N_1{}^{[k]}}.$$

We proceed by induction on k:
$(\mathbf{k = 0})$ The c-net $N_1{}^{[0]}$ consists only of the initial marking of N_1 and thus we have:

$h'_T{}^{[0]} = \emptyset = h_T{}^{[0]}$,
$h'_S{}^{[0]}(s_1, \langle\emptyset, s\rangle) = g_S(s_1, s) = h_S{}^{[0]}(s_1, \langle\emptyset, s\rangle)$, for all $s_1 \in S_1{}^{[0]} = m_1$ and $s \in S$.

$(\mathbf{k \to k+1})$ For all $t_1 \in T^{[k+1]}$, with $depth(t_1) = k+1$ we distinguish two cases:

- If $g_T(t_1) = \bot$ then necessarily
  $h'_T{}^{[k+1]}(t_1) = \bot$ and $\mu h_S{}^{[k+1]}(t_1{}^\bullet) = \emptyset$,
  thus $h'^{[k+1]}$ coincides with $h^{[k+1]}$, on t_1 and its post-set.

- If $g_T(t_1) = t$ then

$$h'_T{}^{[k+1]}(t_1) = t' = \langle M_p, M_c, t\rangle \in T',$$

  with $M_p = {}^\bullet t' = \mu h'_S({}^\bullet t_1)$ and $M_c = \underline{t'} = \mu h'_S(\underline{t_1})$.[11] By inductive hypothesis, since $depth(s_1) \leq k$ for all $s_1 \in {}^\bullet t_1 \cup \underline{t_1}$, we have that $\mu h_S({}^\bullet t_1) = M_p$ and $\mu h_S(\underline{t_1}) = M_c$. Therefore, by definition of h, $h_T(t_1) = \langle M_p, M_c, t\rangle = h'_T(t_1)$.
  Moreover, for all $s_1 \in t_1{}^\bullet$ and for all $s \in t^\bullet$, again by reasoning on commutativity of the diagram, $h'_S(s_1, \langle t', s\rangle) = g_S(s_1, s) = h_S(s_1, \langle t', s\rangle)$.

□

7 Occurrence c-nets and asymmetric event structures

This section shows that the semantics of semi-weighted c-nets given in terms of occurrence c-nets can be abstracted to an event structure and to a domain semantics. First the existence of a coreflection between **AES** and **O-CN** is proved, substantiating the claim according to which AES's represent a suitable model for giving event based semantics to c-nets. Then the coreflection between **AES** and **Dom**, defined in Section 2, can be exploited to complete the chain of coreflections from **SW-CN** to **Dom**.

Given an occurrence c-net we obtain an AES simply forgetting the places, but remembering the dependency relations that they induce between transitions, namely causality and asymmetric conflict. In the same way a morphism between occurrence c-nets naturally restricts to a morphism between the corresponding AES's.

Definition 29 (from occurrence c-nets to AES's). *Let $\mathcal{E}_a : \mathbf{O\text{-}CN} \to \mathbf{AES}$ be the functor defined as:*

- *for each occurrence c-net N:*

$$\mathcal{E}_a(N) = \langle T, \leq_N, \nearrow_N\rangle;$$

- *for each morphism $h : N_0 \to N_1$:*

[11] Notice that here equality holds since we are working with occurrence c-nets and thus contexts can only have multiplicity 1.

$$\mathcal{E}_a(h : N_0 \to N_1) = h_T.$$

Notice that the induced conflict relation $\#^a$ in the AES $\mathcal{E}_a(N)$ (see Definition 3) coincides with the induced conflict relation in the net N (see Definition 25). Therefore in the following we will confuse the two relations and simply write $\#$ to denote both of them.

Proposition 6 (well-definedness). *$\mathcal{E}_a$ is a well-defined functor.*

Proof. Given any occurrence c-net N, from Definition 24, immediately follows that $\mathcal{E}_a(N)$ is an AES. Moreover if $h : N_0 \to N_1$ is a net morphism, then, by Corollary 3, $\mathcal{E}_a(h) = h_T$ is an AES-morphism. Finally $\mathcal{E}_a$ obviously preserves composition and identities. □

An AES can be identified with a canonical occurrence c-net, via a free construction that mimics Winskel's: for each set of events related in a certain way by causal dependency and asymmetric conflict relations we generate a unique place that induces such kind of relations on the events.

Definition 30 (from AES's to occurrence c-nets). *Let $G = \langle E, \leq, \nearrow \rangle$ be an AES. Then $\mathcal{N}_a(G)$ is the net $N = \langle S, T, F, C, m \rangle$ defined as follows:*

- $m = \left\{ \langle \emptyset, A, B \rangle \mid \begin{array}{l} A, B \subseteq E,\ \forall a \in A.\ \forall b \in B.\ a \nearrow b\ \vee\ a\#b, \\ \forall b, b' \in B.\ b \neq b' \Rightarrow b\#b' \end{array} \right\}$;
- $S = m \cup \left\{ \langle \{e\}, A, B \rangle \mid \begin{array}{l} A, B \subseteq E,\ e \in E,\ \forall x \in A \cup B.\ e < x, \\ \forall a \in A.\ \forall b \in B.\ a \nearrow b\ \vee\ a\#b, \\ \forall b, b' \in B.\ b \neq b' \Rightarrow b\#b' \end{array} \right\}$;
- $T = E$;
- $F = \langle F_{pre}, F_{post} \rangle$, *with*
  $F_{pre} = \{(e, s) \mid s = \langle x, A, B \rangle \in S,\ e \in B\}$,
  $F_{post} = \{(e, s) \mid s = \langle \{e\}, A, B \rangle \in S\}$;
- $C = \{(e, s) \mid s = \langle x, A, B \rangle \in S,\ e \in A\}$.

The transitions of $\mathcal{N}_a(G)$ are simply the events of G, while places are triples of the form $\langle x, A, B \rangle$, with $x, A, B \subseteq E$, and $|x| \leq 1$. A place $\langle x, A, B \rangle$ is a precondition for all the events in B, a context for all the events in A. Moreover, if $x = \{e\}$, such place is a postcondition for e, otherwise if $x = \emptyset$ the place belongs to the initial marking. Therefore each place gives rise to a conflict between each pair of (distinct) events in B and to an asymmetric conflict between each pair of events $a \in A$ and $b \in B$. Figure 6 presents some examples of basic AES's with the corresponding c-nets. The cases of an AES with two events related, respectively, by causal dependency, asymmetric conflict and (immediate symmetric) conflict are considered. Pictorially, an asymmetric conflict $e_0 \nearrow e_1$ is represented by a dotted arrow from e_0 to e_1. Causal dependency is represented, as usual, by plain arrows.

The only unexpected thing could be the fact that to insert a place $\langle x, A, B \rangle$ we require each event in A to be in asymmetric conflict *or in conflict* with each event in B. Therefore the construction adds "new" asymmetric conflicts between all pairs of events that are in conflict, since, in general, we can have $e'_0 \# e'_1$ and $\neg(e'_0 \nearrow e'_1)$, as in the AES Figure 7, because $\#$ is inherited via $<$, while $\nearrow$ is not. This is needed for technical reasons, and, abstracting from the formal details, it is quite natural since, being $\#$ the symmetric conflict relation, we can think that conceptually $t \# t'$ implies $t \nearrow t'$.

The next proposition relates the causal dependency and asymmetric conflict relations of an AES with the corresponding relations of the c-net $\mathcal{N}_a(G)$. In particular it is useful in proving that $\mathcal{N}_a(G)$ is indeed an occurrence c-net.

Lemma 17. *Let $G = \langle E, \leq, \nearrow \rangle$ be an AES and let $\mathcal{N}_a(G)$ be the net $N = \langle S, T, F, C, m \rangle$. Then for all $e, e' \in E$:*

1. $e <_N e'$ *iff* $e < e'$;
2. $e \rightsquigarrow_N e'$ *iff* $e \nearrow e'$ *or* $e \# e'$;
3. $e \nearrow_N e'$ *iff* $e \nearrow e'$ *or* $e \# e'$.

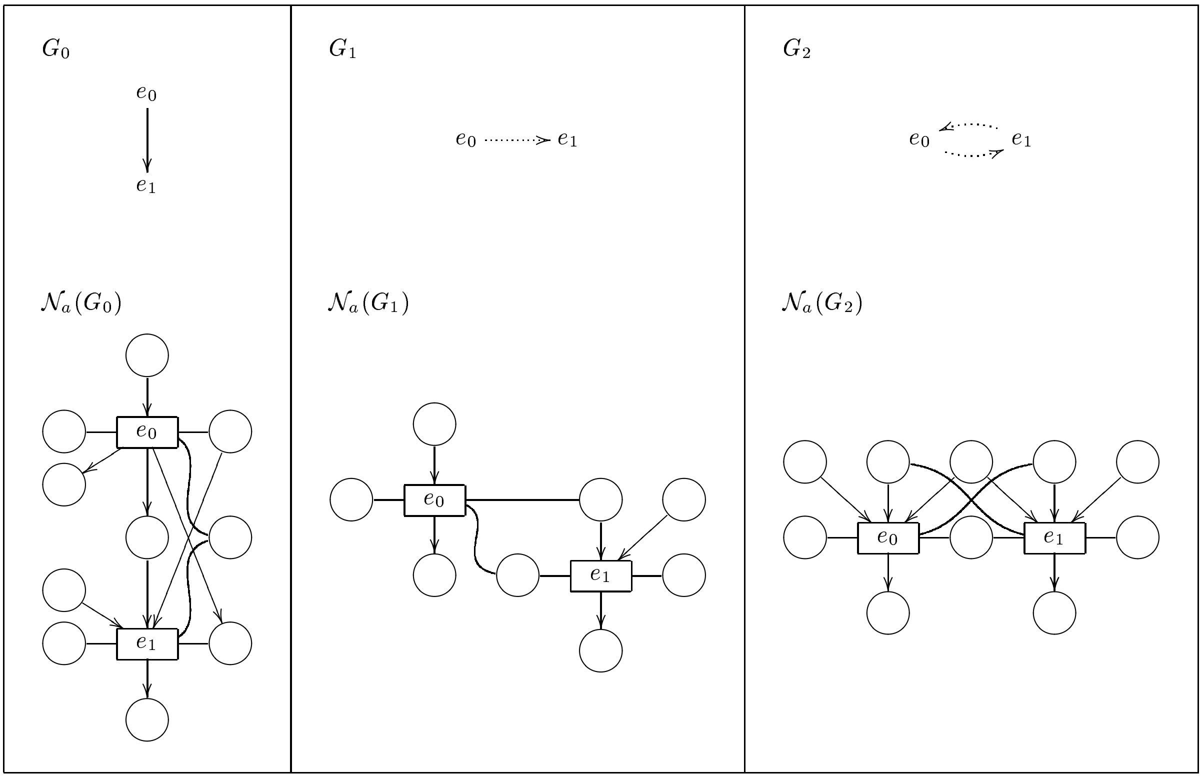


Fig. 6. Three simple AES's and the corresponding occurrence c-nets produced by the functor $\mathcal{N}_a$.

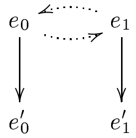


Fig. 7. An AES with two conflictual events e_0' and e_1', not related by asymmetric conflict.

Proof. 1. Let $\prec_N$ and $\prec$ denote the immediate causality relations. If $e \prec_N e'$ then there exists a place $\langle \{e\}, A, B\rangle \in S$ with $e' \in A \cup B$ and thus, by definition of $\mathcal{N}_a$, $e < e'$. Viceversa, if $e < e'$ then $\langle \{e\}, \emptyset, \{e'\}\rangle \in S$ and thus $e \prec_N e'$. Being $<_N$ and $<$ the transitive closures of $\prec_N$ and $\prec$ we have the thesis.

2. If $e \rightsquigarrow_N e'$, then there exists a place $\langle x, A, B\rangle \in S$ with $e \in A \cup B$ and $e' \in B$ and thus $e \nearrow e'$ or $e \# e'$. Viceversa, if $e \nearrow e'$ or $e \# e'$ then $\langle \emptyset, \{e\}, \{e'\}\rangle \in S$ and thus $e \rightsquigarrow_N e'$.

3. Immediate consequence of 1 and 2.

□

As an immediate corollary we have:

Corollary 4. *Let $G = \langle E, \leq, \nearrow\rangle$ be an* AES*. Then $\mathcal{N}_a(G) = N = \langle S, T, F, C, m\rangle$ is an occurrence c-net.*

Proof. The causal relation $<_N$, by Lemma 17, inherits the necessary properties from those of G. Moreover $\nearrow_N = \nearrow \cup \#$ and thus for all $e \in T$, since $\lfloor e\rfloor_N = \lfloor e\rfloor_G$ and $\lfloor e\rfloor_G$ is conflict-free, $(\nearrow_N)_{\lfloor e\rfloor_N} = \nearrow_{\lfloor e\rfloor_G}$ is acyclic. □

Let $G = \langle E, \leq, \nearrow\rangle$ be an AES. For $e \in E$, we define the set of *consequences* $\lceil\{e\}\rceil$, as follows (considering the singleton $\{e\}$ instead of e itself will simplify the notation later).

$$\lceil\{e\}\rceil = \{e' \in E \mid e < e'\}.$$

This function is extended also to the empty set, by $\lceil\emptyset\rceil = E$. We use the same notation for occurrence c-nets, referring to the underlying AES.

The next technical lemma gives a property of morphisms between occurrence c-nets which will be useful in the proof of the coreflection result.

Lemma 18. *Let N_0 and N_1 be occurrence c-nets and let $h : N_0 \to N_1$ be a morphism. For all $s_0 \in S_0$ and $s_1 \in S_1$, if $h_S(s_0, s_1)$ then*

1. *$h_T({}^\bullet s_0) = {}^\bullet s_1$;*
2. *$s_0{}^\bullet = h_T^{-1}(s_1{}^\bullet) \cap \lceil {}^\bullet s_0\rceil$;*
3. *$\underline{s_0} = h_T^{-1}(\underline{s_1}) \cap \lceil {}^\bullet s_0\rceil$.*

Proof. Let $s_0 \in S_0$ and $s_1 \in S_1$ such that $h_S(s_0, s_1)$.

1. If ${}^\bullet s_0 = \emptyset$, i.e., $s_0 \in m_0$ then $s_1 \in m_1$ and thus ${}^\bullet s_1 = \emptyset = h_T({}^\bullet s_0)$. Otherwise, let ${}^\bullet s_0 = \{t_0\}$.[12] Therefore $h_T(t_0) = t_1$ is defined (see the remark after Definition 20) and $s_1 \in t_1{}^\bullet$. Thus ${}^\bullet s_1 = \{t_1\} = h_T({}^\bullet s_0)$.

2. Let $t_0 \in s_0{}^\bullet$, i.e., $s_0 \in {}^\bullet t_0$. Since $h_S(s_0, s_1)$, we have that $h_T(t_0) = t_1$ is defined and $s_1 \in {}^\bullet t_1$. Thus $t_0 \in h_T^{-1}(s_1{}^\bullet) \cap \lceil {}^\bullet s_0\rceil$.
For the converse inclusion, let $t_0 \in h_T^{-1}(s_1{}^\bullet) \cap \lceil {}^\bullet s_0\rceil$. Then $s_1 \in {}^\bullet h_T(t_0)$ and thus there is $s_0' \in {}^\bullet t_0$ such that $h_S(s_0', s_1)$. Now, reasoning as in Lemma 16.(1), we conclude that s_0' and s_0 necessarily coincide, otherwise they would be in the post-condition of conflicting transitions and thus, since $t_0 \in \lceil {}^\bullet s_0\rceil$, we would have $t_0 \# t_0$.

3. Analogous to 2.

□

Let $G = \langle E, \leq, \nearrow\rangle$ be an AES. By Lemma 17, $\mathcal{E}_a(\mathcal{N}_a(G)) = \langle E, \leq, \nearrow'\rangle$, where $\nearrow' = \nearrow \cup \#$. Therefore the identity on events

$$\eta_G : G \to \mathcal{E}_a(\mathcal{N}_a(G))$$

[12] There is a unique transition generating s_0, since N_0 is an occurrence c-net.

defined by $\eta_G(e) = e$, for all $e \in E$, is an AES-morphism. In fact, $\mathcal{E}_a(\mathcal{N}_a(G))$ differs from G only for the asymmetric conflict relation and it is easy to see that morphism conditions are satisfied. Moreover $\eta_G^{-1} : \mathcal{E}_a(\mathcal{N}_a(G)) \to G$, again defined as identity on events is clearly a morphism, and η_G and η_G^{-1} are inverse to each other. Therefore η_G is an isomorphism.

We are now ready to prove that $\mathcal{N}_a$ extends to a functor from **AES** to **O-CN**, which is left adjoint to $\mathcal{E}_a$ (with unit η_G). More precisely they establish a coreflection from **AES** to **O-CN**.

Theorem 6 (coreflection between O-CN and AES). $\mathcal{N}_a \dashv \mathcal{E}_a$

Proof. Let $G = \langle E, \leq, \nearrow \rangle$ be an AES, let $\mathcal{N}_a(G) = \langle S, T, F, C, m \rangle$ be as in Definition 30, and let $\mathcal{E}_a(\mathcal{N}_a(G)) = \langle E, \leq, \nearrow' \rangle$, where $\nearrow' = \nearrow \cup \#$. We have to show that for any occurrence c-net N_0 and for any morphism $g : G \to \mathcal{E}_a(N_0)$ there exists a unique morphism $h : \mathcal{N}_a(G) \to N_0$, such that the following diagram commutes:

$$\begin{tikzcd} G \arrow[r, "\eta_G"] \arrow[dr, "g"'] & \mathcal{E}_a(\mathcal{N}_a(G)) \arrow[d, dotted, "\mathcal{E}_a(h)"] \\ & \mathcal{E}_a(N_0) \end{tikzcd}$$

The behaviour of h on transitions is determined immediately by g:

$$h_T = g.$$

Therefore we must show that a multirelation $h_S : S \to S_0$ such that $\langle h_T, h_S \rangle$ is a morphism exists and is uniquely determined by h_T.

Existence
Let us define h_S in such a way it satisfies the conditions of Lemma 18, specialized to the net $\mathcal{N}_a(G)$, that is, for all $s = \langle x, A, B \rangle \in S$ and $s_0 \in S_0$:

$$\begin{aligned} h_S(s, s_0) \quad \text{iff} \quad & ((x = \emptyset \ \wedge \ s_0 \in m_0) \ \vee \ (x = \{t\} \ \wedge \ s_0 \in h_T(t)^\bullet)) \\ & \wedge\ B = h_T^{-1}(s_0{}^\bullet) \cap \lceil x \rceil \\ & \wedge\ A = h_T^{-1}(\underline{s_0}) \cap \lceil x \rceil \end{aligned}$$

To prove that the pair $h = \langle h_T, h_S \rangle$ is indeed a morphism, let us verify the conditions on the preservation of the initial marking and of the pre-set, post-set and context of transitions.

First observe that $\mu h_S(m) = m_0$. In fact, if $s = \langle x, A, B \rangle \in m$ and $h_S(s, s_0)$ then $x = \emptyset$ and thus, by definition of h_S, $s_0 \in m_0$. Viceversa, let $s_0 \in m_0$ and let

$$A = h_T^{-1}(\underline{s_0}) \qquad \text{and} \qquad B = h_T^{-1}(s_0{}^\bullet)$$

Since $t_0 \# t_0'$ for all $t_0, t_0' \in s_0{}^\bullet$ and $t_0 \nearrow t_0'$ for all $t_0 \in \underline{s_0}$, $t_0' \in s_0{}^\bullet$, by definition of AES-morphism, $t \# t'$ for all $t, t' \in B$, and $t \nearrow t'$ or $t \# t'$ for all $t \in A$ and $t' \in B$. Hence there is a place $s = \langle \emptyset, A, B \rangle \in m$ and $h_S(s, s_0)$.

Now, let $t \in T$ be any transition, such that $h_T(t)$ is defined. Then

- $\mu h_S({}^\bullet t) = {}^\bullet h_T(t)$.
  In fact, let $s = \langle x, A, B \rangle \in {}^\bullet t$, that is $t \in B$, and let $h_S(s, s_0)$. Then, by definition of h_S, $h_T(t) \in s_0{}^\bullet$, or equivalently $s_0 \in {}^\bullet h_T(t)$. For the converse inclusion, let $s_0 \in {}^\bullet h_T(t)$ and let $x = h_T^{-1}({}^\bullet s_0) \cap \lfloor t \rfloor$. By Lemma 16, $|\ x\ | \leq 1$ (more precisely $x = \emptyset$ if $s_0 \in m_0$, otherwise, x contains the unique $t' \leq t$, such that $h_T(t') = t_0$, with ${}^\bullet s_0 = \{t_0\}$). Consider

  $$A = h_T^{-1}(\underline{s_0}) \cap \lceil x \rceil \quad \text{and} \quad B = h_T^{-1}(s_0{}^\bullet) \cap \lceil x \rceil.$$

  Since $t_0 \# t_0'$ for all $t_0, t_0' \in s_0{}^\bullet$ and $t_0 \nearrow t_0'$ for all $t_0 \in \underline{s_0}$, $t_0' \in s_0{}^\bullet$, as in the previous case, we have that $s = \langle x, A, B \rangle \in S$ is a place such that $h_S(s, s_0)$. Clearly $t \in \lceil x \rceil$, thus $t \in B$ and therefore $s \in {}^\bullet t$ and $s_0 \in \mu h_S({}^\bullet t)$.

– $\mu h_S(\underline{t}) = \underline{h_T(t)}$.
  Analogous to the previous case.

– $\mu h_S(t^\bullet) = h_T(t)^\bullet$.
  If $s = \langle x, A, B\rangle \in t^\bullet$, that is $x = \{t\}$, and $h_S(s, s_0)$, then, by definition of h_S, we have $s_0 \in h_T(t)^\bullet$. For the converse, let $s_0 \in h_T(t)^\bullet$. As above, consider

$$A = h_T^{-1}(\underline{s_0}) \cap \lceil\{t\}\rceil \quad \text{and} \quad B = h_T^{-1}(s_0{}^\bullet) \cap \lceil\{t\}\rceil.$$

  Then $s = \langle\{t\}, A, B\rangle \in t^\bullet$ and, by definition of h_S, we have $h_S(s, s_0)$.

Finally, if $h_T(t)$ is not defined, then the definition of h_S implies that $\mu h_S({}^\bullet t) = \mu h_S(\underline{t}) = \mu h_S(t^\bullet) = \emptyset$. This concludes the proof that h is a morphism.

Uniqueness
The multirelation h_S such that $\langle h_T, h_S\rangle$ is a c-net morphism is unique essentially because it must satisfy the conditions of Lemma 18. More precisely, if $h'_S : S \to S_0$ is another multirelation, such that $\langle h_T, h'_S\rangle$ is a morphism and $h'_S(s, s_0)$ then necessarily by Lemma 18, $h_S(s, s_0)$. Conversely, let $h_S(s, s_0)$, with $s = \langle x, A, B\rangle$. Then, if $x = \emptyset$, by properties of net morphisms, $s_0 \in m_0$. Therefore there must be $s' \in m$ such that $h'_S(s', s_0)$. But, by Lemma 18 and definition of h_S, $\underline{s'} = h_T^{-1}(\underline{s_0}) = A$ and similarly $s'^\bullet = h_T^{-1}(s_0{}^\bullet) = B$. Therefore $s' = \langle\emptyset, A, B\rangle = s$ and thus $h'_S(s, s_0)$. An analogous reasoning allow us to conclude in the case $x = \{t\}$. □

We know by the previous theorem that $\mathcal{N}_a$ extends to a functor from **AES** to **O-CN**. The behaviour of $\mathcal{N}_a$ on morphisms is suggested by the proof of the theorem. Let $h : G_0 \to G_1$, be an AES-morphism and let $\mathcal{N}_a(G_i) = \langle S_i, T_i, F_i, C_i, m_i\rangle$ for $i \in \{0, 1\}$. Then $\mathcal{N}_a(h) = \langle h, h_S\rangle$, with h_S defined as follows:

– for all places $\langle\emptyset, A_1, B_1\rangle$

$$h_S(\langle\emptyset, h^{-1}(A_1), h^{-1}(B_1)\rangle, \langle\emptyset, A_1, B_1\rangle)$$

– for all $e_0 \in T_0$ such that $h_T(e_0) = e_1$ and for all places $\langle\{e_1\}, A_1, B_1\rangle$

$$h_S(\langle\{e_0\}, h^{-1}(A_1) \cap \lceil e_0\rceil, h^{-1}(B_1) \cap \lceil e_0\rceil\rangle, \langle\{e_1\}, A_1, B_1\rangle)$$

Finally, notice that, as an easy extension, the equivalence between **PES** and **Dom** (see Subsection 3.1) can be used to "translate" the domain semantics of semi-weighted c-nets into a traditional event structure semantics. This completes the following chain of coreflections between **SW-CN** and **PES**

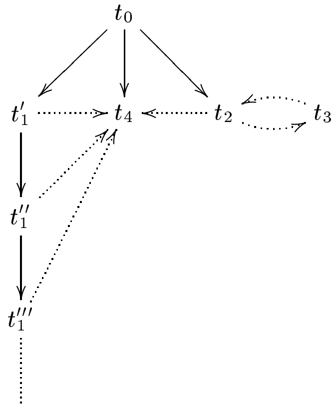
$$\begin{tikzcd} \mathbf{SW\text{-}CN} \arrow[r, shift right=1.5, "\mathcal{U}_a"'] \arrow[r, phantom, "\perp"] & \mathbf{O\text{-}CN} \arrow[l, hook', shift right=1.5, "\mathcal{I}_O"'] \arrow[r, shift right=1.5, "\mathcal{E}_a"'] \arrow[r, phantom, "\perp"] & \mathbf{AES} \arrow[l, shift right=1.5, "\mathcal{N}_a"'] \arrow[r, shift right=1.5, "\mathcal{L}_a"'] \arrow[r, phantom, "\perp"] & \mathbf{Dom} \arrow[l, shift right=1.5, "\mathcal{P}_a"'] \arrow[r, shift right=1.5, "\mathcal{P}"'] \arrow[r, phantom, "\sim"] & \mathbf{PES} \arrow[l, shift right=1.5, "\mathcal{L}"'] \end{tikzcd}$$

Figure 8 shows (part of) the AES, the domain and the PES associated to the c-net of Figure 5. Although (for the sake of readability) not explicitly drawn, in the PES all the "copies" of t_4, namely the events t_4^x are in conflict.

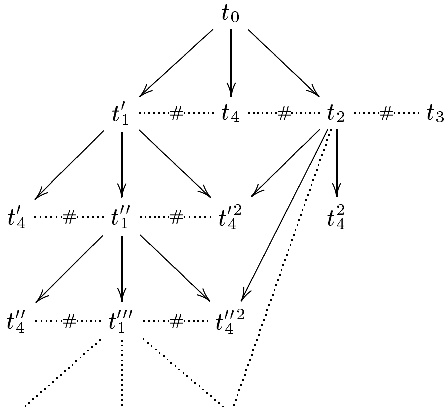
We remark that the PES semantics is obtained from the AES semantics by introducing an event for each possible different history of events in the AES. For instance, the PES semantics of the net N_3 in Figure 9 is given by P, where e'_1 represents the firing of the transition t_1 by itself, with an empty history, and e''_1 the firing of the transition t_1 after t_0. Obviously the AES semantics is finer than the PES semantics, or in other words the translation from **AES** to **PES** causes a loss of information. For example, the nets N_3 and N'_3 in Figure 9 have the same PES semantics, but different AES semantics.

8 Processes and unfolding

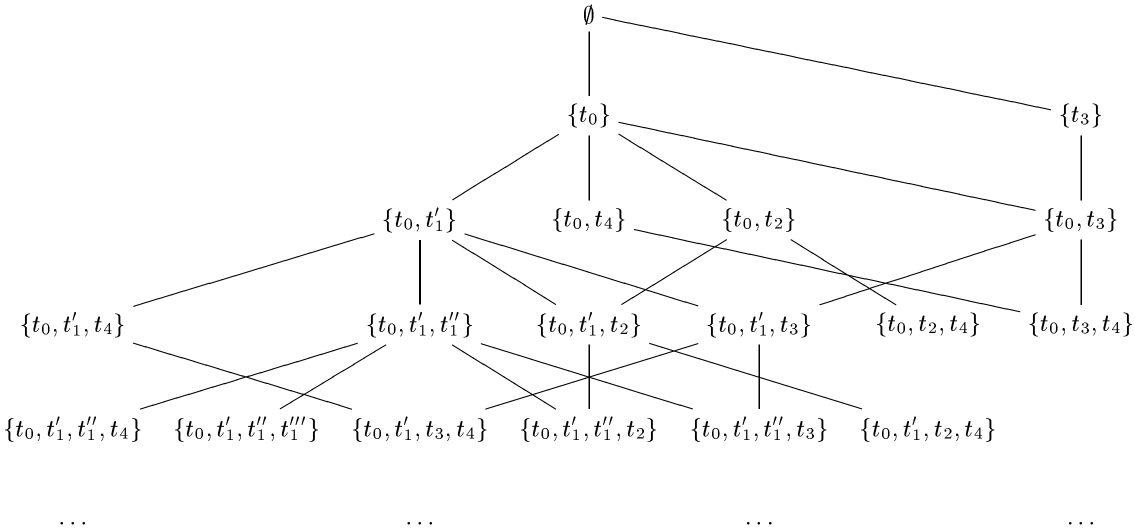
The notion of occurrence c-net introduced in Section 5 naturally suggests a notion of nondeterministic process for c-nets, which can be defined as an occurrence c-net with a morphism (mapping places into places and total on transitions) to the original net. Deterministic c-net processes can then be defined



(a) $\mathcal{E}_a(\mathcal{U}_a(N))$

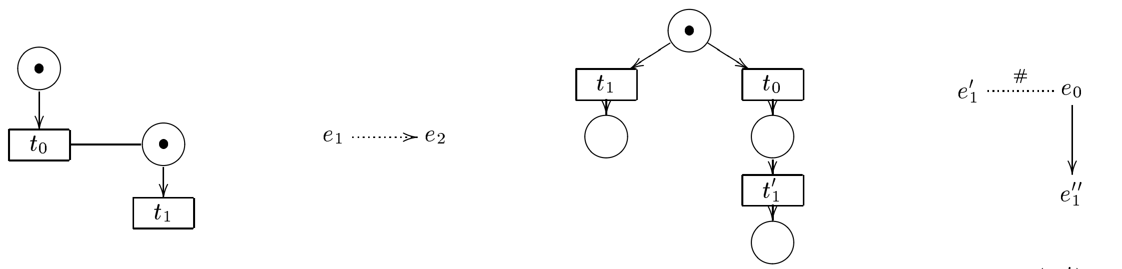


(c) $\mathcal{P}(\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N))))$



(b) $\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N)))$

Fig. 8. The (a) AES (b) domain and (c) PES for the c-net N of Figure 5



N_3 $\quad$ $G = \mathcal{E}_a(N_3)$ $\quad$ N'_3 $\quad$ $P = \mathcal{E}_a(N'_3) = \mathcal{P}(\mathcal{L}_a(\mathcal{E}_a(N_3)))$

Fig. 9. AES semantics is finer than PES semantics.

as particular nondeterministic processes such that the underlying occurrence c-net satisfies a further conflict-freeness requirement. Interestingly, the resulting notion of deterministic process turns out to coincide with those proposed by other authors [36, 34, 17, 38]. The PhD thesis [7] introduces processes for nets with read and inhibitor arcs, which, restricted to the subclass of nets without inhibitor arcs, still coincide with ours. Furthermore it is worth recalling that the stress on the necessity of using an additional relation of "weak-causality" to be able to fully express the causal structure of net computations in the presence of read or inhibitor arcs can be found already in [20, 21]. [13]

The papers [34, 17, 38] extend the theory of concatenable processes of traditional nets [15] to c-nets, by showing that the concatenable processes of a c-net N form the arrows of a symmetric monoidal category $\mathbf{CP}[N]$, where objects are the elements of the free commutative monoid over the set of places (multisets of places). In particular, in [17] a purely algebraic characterization of such category is given.

This section, relying on the notion of concatenable c-net process and exploiting the chain of coreflections leading from **SW-CN** to **Dom**, establishes a close relationship between process and unfolding semantics for c-nets. More precisely, we generalize to c-nets (in the semi-weighted case) an analogous result proved in [25] for traditional nets, stating that the domain $\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N)))$ associated to a semi-weighted net N coincides with the completion of the preorder obtained as the comma category of $\mathbf{CP}[N]$ over the initial marking. Roughly speaking, the result says that the domain obtained via the unfolding of a c-net can be equivalently described as the collection of the deterministic processes of the net, ordered by prefix.

8.1 Contextual nets processes

A process of a c-net N can be naturally defined as an occurrence c-net N_π, together with a morphism π to the original net. In fact, since morphisms preserve the token game, π maps computations of N_π into computations of N in such a way that the process can be seen as a representative of a set of possible computations of N. The occurrence c-net N_π makes explicit the causal structure of such computations since each transition is fired at most once and each place is filled at most with one token during each computation. In this way (as it happens in the unfolding) transitions and places of N_π can be thought of, respectively, as instances of firing of transitions and tokens in places of the original net. Actually, to allow for such an interpretation, some further restrictions have to be imposed on the morphism π, namely it must map places into places (rather than into multisets of places) and it must be total on transitions.

Besides "marked processes", representing computations of the net starting from the initial marking, we will introduce also "unmarked processes", representing computations starting from a generic marking. This is needed to be able to define a meaningful notion of concatenation between processes.

Definition 31 (process). *A* marked process *of a c-net $N = \langle S, T, F, C, m\rangle$ is a mapping $\pi : N_\pi \to N$, where N_π is an occurrence c-net and π is a strong c-net morphism, namely a c-net morphism such that π_T is total and π_S maps places into places. The process is called* discrete *if N_π has no transitions.*

An unmarked process *of N is defined in the same way, where the mapping π is an "unmarked morphism", namely π is not required to preserve the initial marking (it satisfies all conditions of Definition 20, but (1)).*

Equivalently, if we denote by $\mathbf{CN}^*$ the subcategory of $\mathbf{CN}$ where the arrows are *strong c-net* morphisms, the processes of N can be seen as objects of the comma category $(\mathbf{O\text{-}CN} \downarrow N)$ in $\mathbf{CN}^*$. This gives also the (obvious) notion of isomorphism between processes, which is a morphism between the underlying occurrence nets "consistent" with the mappings to the original net. Analogous definitions can be given also for the unmarked processes of a net N.[14]

[13] A different notion of enabling allowing for the simultaneous firing of weakly dependent transitions is used in [21], making difficult a complete direct comparison. For the same reason, although "syntactically" the processes of [36] coincide with ours, they are intended to represent the same firing sequences, but different step sequences.

[14] It is worth remarking that if we want each truly concurrent computation of the net N to be represented by at most one configuration of the nondeterministic process, an additional constraint must be imposed on π, requiring that ${}^\bullet t_1 = {}^\bullet t_2$, $\underline{t_1} = \underline{t_2}$ and $\pi(t_1) = \pi(t_2)$ implies $t_1 = t_2$, as in [37]. However, the two notions of process collapse when we restrict to deterministic processes which are the focus of this section.

A deterministic process represents a set of computations which differ only for the order in which independent transitions are fired. In our setting a deterministic process is thus defined as a process such that, in the underlying occurrence net, the transitive closure of asymmetric conflict is a finitary partial order, in such a way that all transitions can be fired in a single computation of the net. Deterministic occurrence c-nets will be always denoted by O, possibly with subscripts.

Definition 32 (deterministic occurrence c-net). *An occurrence c-net O is called* deterministic *if the asymmetric conflict $\nearrow_O$ is acyclic and well-founded.*

Equivalently, one could have asked the transitive closure of the asymmetric conflict relation $(\nearrow_O)^*$ to be a partial order, such that for each transition t in O, the set $\{t' \mid t'(\nearrow_O)^* t\}$ is finite. Alternatively, it can be easily seen that an occurrence c-net is deterministic if and only if the corresponding AES is conflict free.

We denote by $\min(O)$ and $\max(O)$ the sets of minimal and maximal places of O with respect to the partial order $\leq_O$.

Definition 33 (deterministic process). *A (marked or unmarked) process π is called* deterministic *if the occurrence c-net O_π is deterministic. The process is* finite *if the set of transitions in O_π is finite. In this case, we denote by $\min(\pi)$ and $\max(\pi)$ the sets $\min(O_\pi)$ and $\max(O_\pi)$, respectively. Moreover we denote with ${}^\bullet\pi$ and $\pi^\bullet$ the multisets $\mu\pi(\min(\pi))$ and $\mu\pi(\max(\pi))$, called respectively the* source *and the* target *of π.*

Clearly, in the case of a marked process π of a c-net N, the marking ${}^\bullet\pi$ coincides with the initial marking of N.

8.2 Concatenable processes

As in [17, 38] a notion of concatenable process for contextual nets, endowed with a notion of sequential (and parallel) composition, can be easily defined, generalizing the concatenable processes of [15]. Obviously a meaningful operation of sequential composition can be defined only on the unmarked processes of a c-net. In order to properly define the operation of concatenation of processes, we need to impose a suitable ordering over the places in $\min(\pi)$ and $\max(\pi)$ for each process π. Such ordering allows to distinguish among "interface" places of O_π which are mapped to the same place of the original net, which is essential to make sequential composition consistent with the causal dependencies.

Definition 34. *Let A and B be sets and let $f : A \to B$ be a function. An f-*indexed ordering *is a family $\alpha = \{\alpha_b \mid b \in B\}$ of bijections $\alpha_b : f^{-1}(b) \to [|f^{-1}(b)|]$, where $[i]$ denotes the subset $\{1, \ldots, i\}$ of $\mathbb{N}$, and $f^{-1}(b) = \{a \in A \mid f(a) = b\}$.*

The f-indexed ordering α will be often identified with the function from A to $\mathbb{N}$ it naturally induces (formally defined as $\bigcup_{b \in B} \alpha_b$).

Definition 35 (concatenable process). *A* concatenable process *of a c-net N is a triple $\delta = \langle \mu, \pi, \nu \rangle$, where*

- *π is a finite deterministic unmarked process of N;*
- *μ is π-indexed ordering of $\min(\pi)$;*
- *ν is π-indexed ordering of $\max(\pi)$;*

Two concatenable processes $\delta_1 = \langle \mu_1, \pi_1, \nu_1 \rangle$ and $\delta_2 = \langle \mu_2, \pi_2, \nu_2 \rangle$ of a c-net N are *isomorphic* if there exists an isomorphism of processes $f : \pi_1 \to \pi_2$, consistent with the decorations, i.e., such that $\mu_2(f_S(s_1)) = \mu_1(s_1)$ for each $s_1 \in \min(\pi_1)$ and $\nu_2(f_S(s_1)) = \nu_1(s_1)$ for each $s_1 \in \max(\pi_1)$. An isomorphism class of processes is called *(abstract) concatenable process* and denoted by $[\delta]$, where δ is a member of that class. In the following we will often omit the word "abstract" and write δ to denote the corresponding equivalence class.

The operation of sequential composition on concatenable processes is defined in the natural way. Given two concatenable processes $\langle \mu_1, \pi_1, \nu_1 \rangle$ and $\langle \mu_2, \pi_2, \nu_2 \rangle$, such that ${\pi_1}^\bullet = {}^\bullet\pi_2$ their concatenation is defined as the process obtained by gluing the maximal places of π_1 and the minimal places of π_2 according to the ordering of such places.

Definition 36 (sequential composition). *Let $\delta_1 = \langle \mu_1, \pi_1, \nu_1 \rangle$ and $\delta_2 = \langle \mu_2, \pi_2, \nu_2 \rangle$ be two concatenable processes of a c-net N such that ${\pi_1}^\bullet = {}^\bullet\pi_2$. Suppose $T_1 \cap T_2 = \emptyset$ and $S_1 \cap S_2 = \max(\pi_1) = \min(\pi_2)$, with $\pi_1(s) = \pi_2(s)$ and $\nu_1(s) = \mu_2(s)$ for each $s \in S_1 \cap S_2$. In words δ_1 and δ_2 overlap only on $\max(\pi_1) = \min(\pi_2)$, and on such places their labelling on the original net and the ordering coincide. Then their concatenation $\delta_1; \delta_2$ is the concatenable process $\delta = \langle \mu_1, \pi, \nu_2 \rangle$, where the process π is the (componentwise) union of π_1 and π_2*

It is easy to see that concatenation induces a well-defined operation of sequential composition between abstract processes. In particular, if $[\delta_1]$ and $[\delta_2]$ are abstract concatenable processes such that ${\delta_1}^\bullet = {}^\bullet\delta_2$ then we can always find $\delta_2' \in [\delta_2]$ such that $\delta_1; \delta_2'$ is defined. Moreover the result of the composition at abstract level, namely $[\delta_1; \delta_2']$, does not depend on the particular choice of the representatives.

Definition 37 (category of concatenable processes). *Let N be a c-net. The* category of (abstract) concatenable processes *of N, denoted by $\mathbf{CP}[N]$, is defined as follows. Objects are multisets of places of N, namely elements of μS. Each (abstract) concatenable process $[\langle \mu, \pi, \nu \rangle]$ of N is an arrow from ${}^\bullet\pi$ to $\pi^\bullet$.*

One could also define a tensor operation $\otimes$, modeling parallel composition of processes, making the category $\mathbf{CP}[N]$ a symmetric monoidal category (the symmetries being the discrete processes). Since such operation is not relevant for our present aim, we refer the interested reader to [17, 38].

8.3 Relating processes and unfolding

Let $N = \langle S, T, F, C, m \rangle$ be a c-net and consider the comma category $(m \downarrow \mathbf{CP}[N])$. The objects of such category are concatenable processes of N starting from the initial marking. An arrow exists from a process δ_1 to δ_2 if the second one can be obtained by concatenating the first one with a third process δ. This can be interpreted as a kind of prefix ordering.

Lemma 19. *Let $N = \langle S, T, F, C, m \rangle$ be a c-net. Then the comma category $(m \downarrow \mathbf{CP}[N])$ is a preorder.*

Proof. Let $\delta_i : m \to M_i$ ($i \in \{1, 2\}$) be two objects in $(m \downarrow \mathbf{CP}[N])$, and suppose there are two arrows $\delta', \delta'' : \delta_1 \to \delta_2$. By definition of comma category $\delta_1; \delta' = \delta_1; \delta'' = \delta_2$, which, by definition of sequential composition, easily implies $\delta' = \delta''$. □

In the sequel the preorder relation over $(m \downarrow \mathbf{CP}[N])$ (induced by sequential composition) will be denoted by $\lesssim_N$ or simply by $\lesssim$, when the net N is clear from the context. Therefore $\delta_1 \lesssim \delta_2$ if there exists δ such that $\delta_1; \delta = \delta_2$.

We provide an alternative characterization of the preorder relation $\lesssim_N$ which will be useful in the sequel. It essentially formalizes the intuition given above, according to which the preorder on $(m \downarrow \mathbf{CP}[N])$ is a generalization of the prefix relation. First, we need to introduce the notion of left-injection for processes.

Definition 38 (left injection). *Let $\delta_i : m \to M_i$ ($i \in \{1, 2\}$) be two objects in $(m \downarrow \mathbf{CP}[N])$, with $\delta_i = \langle \mu_i, \pi_i, \nu_i \rangle$. A* left injection *$\iota : \delta_1 \to \delta_2$ is a morphism of marked processes $\iota : \pi_1 \to \pi_2$, such that*

1. *ι is consistent with the indexing of minimal places, namely $\mu_1(s) = \mu_2(\iota(s))$ for all $s \in \min(\pi_1)$;*
2. *ι is "rigid" on transitions, namely for t_2' in O_{π_2} and t_1 in O_{π_1}, if $t_2' \nearrow \iota(t_1)$ then $t_2' = \iota(t_1')$ for some t_1' in O_{π_1}.*

The name "injection" is justified by the fact that a morphism ι between marked deterministic processes (being a morphism between the underlying occurrence c-nets) is injective on places and transitions, as it can be shown easily by using the properties of (occurrence) c-nets morphisms proved in Section 5. The word "left" is instead related to the requirement of consistency with the decoration of the minimal items. Finally, the rigidity of the morphism ensures that δ_2 does not extend δ_1 with transitions inhibited in δ_1.

Lemma 20. *Let $\delta_i : m \to M_i$ ($i \in \{1, 2\}$) be two objects in $(m \downarrow \mathbf{CP}[N])$, with $\delta_i = \langle \mu_i, \pi_i, \nu_i \rangle$. Then*

$$\delta_1 \lesssim \delta_2 \quad \textit{iff} \quad \textit{there exists a left injection } \iota : \delta_1 \to \delta_2.$$

Proof. ($\Rightarrow$) Let $\delta_1 \lesssim \delta_2$, namely $\delta_2 = \delta_1; \delta$ for some process $\delta = \langle \mu, \pi, \nu \rangle$. Without loss of generality, we can imagine that π_2 is obtained as the componentwise union of π_1 and π and this immediately gives a morphism of marked processes (the inclusion) $\iota : \pi_1 \to \pi_2$, consistent with the indexing of minimal places. To conclude it remains only to show that ι is rigid. Suppose that $t_2' \nearrow \iota(t_1)$ for some transitions t_1 in O_{π_1} and t_2' in O_{π_2}. Since the process π_2 is deterministic, t_2' and $\iota(t_1)$ cannot be in conflict and thus, by Definition 23, it must be $(t_2'^{\bullet} \cup \underline{t_2'}) \cap {}^{\bullet}\iota(t_1) \neq \emptyset$. Since t_2' uses as context or postcondition a place which is not maximal in π_1, necessarily t_2' is in O_{π_1}, otherwise it could not be added by concatenating π to π_1. Therefore we conclude that ι is rigid.

($\Leftarrow$) Let $\iota : \delta_1 \to \delta_2$ be a left injection. We can suppose without loss of generality that O_{π_1} is a subnet of O_{π_2}, in such a way that ι is the inclusion and $\mu_1 = \mu_2$. Let O_π be the net $(O_{\pi_2} \setminus O_{\pi_1}) \cup \max(O_{\pi_1})$, where difference and union are defined componentwise. More precisely $O_\pi = \langle S, T, F, C \rangle$, with:

- $S = (S_2 \setminus S_1) \cup \max(\pi_1)$
- $T = T_2 \setminus T_1$
- the relations F and C are the restrictions of F_2 and C_2 to T.

It is easy to see that O_π is a well-defined occurrence c-net and $\min(O_\pi) = \max(O_{\pi_1})$. In particular, the fact that F is well-defined namely that if $t \in T$ then ${}^{\bullet}t, t^{\bullet} \subseteq S$ immediately derives from the fact that the inclusion ι is a morphism of deterministic occurrence c-nets. Instead the well-definedness of C is related to the fact that the injection is rigid. In fact, let $s \in \underline{t}$ for $t \in T$ and suppose that $s \notin S$. Therefore $s \in {}^{\bullet}t_1$, for some $t_1 \in T_1$ and thus $t \nearrow t_1$, which, by rigidity, implies $t \in T_1$, contradicting $t \in T$.

Therefore if we denote by δ the concatenable process $\langle \nu_1, \pi, \nu_2 \rangle$, then $\delta_1; \delta = \delta_2$, and thus $\delta_1 \lesssim \delta_2$. $\square$

We can now show that the ideal completion of the preorder $(m \downarrow \mathbf{CP}[N])$ is isomorphic to the domain obtained from the unfolding of the net N, namely $\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N)))$. Besides exploiting the characterization of the preorder relation on $(m \downarrow \mathbf{CP}[N])$ given above, the result strongly relies on the description of the unfolding construction as chain of adjunctions.

First, it is worth recalling some definitions and results on the ideal completion of (pre)orders.

Definition 39 (ideal). *Let P be a preorder. An* ideal *of P is a subset $S \subseteq P$, directed and downward closed (namely $S = \bigcup\{\downarrow x \mid x \in S\}$). The set of ideals of P, ordered by subset inclusion is denoted by $\mathsf{Idl}(P)$.*

Given a preorder P, the partial order $\mathsf{Idl}(P)$ is an algebraic CPO, with compact elements $\mathsf{K}(\mathsf{Idl}(P)) = \{\downarrow p \mid p \in P\}$. Moreover $\mathsf{Idl}(P) \simeq \mathsf{Idl}(P/_\equiv)$, where $P/_\equiv$ is the partial order induced by the preorder P. Finally, recall that if D is an algebraic CPO, then $\mathsf{Idl}(\mathsf{K}(D)) \simeq D$.

Lemma 21. *Let P_1 and P_2 be preorders and let $f : P_1 \to P_2$ be a surjective function such that $p_1 \sqsubseteq p_1'$ iff $f(p_1) \sqsubseteq f(p_1')$. Then the function $f^* : \mathsf{Idl}(P_1) \to \mathsf{Idl}(P_2)$, defined by $f^*(I) = \{f(x) \mid x \in I\}$, for $I \in \mathsf{Idl}(P_1)$, is an isomorphism of partial orders.*

Proof. The function f^* is surjective since for every ideal $I_2 \in \mathsf{Idl}(P_2)$ can be easily proved that $f^{-1}(I_2)$ is an ideal and $f^*(f^{-1}(I_2)) = I_2$ by surjectivity of f.. Moreover, notice that if $I_1, I_1' \in \mathsf{Idl}(P_1)$ are two ideals then $I_1 \subseteq I_1'$ if and only if $f^*(I_1) \subseteq f^*(I_1')$. The left implication is obvious. For the right one, assume $f^*(I_1) \subseteq f^*(I_1')$. Then observe that if $x \in I_1$ then $f(x) \in f^*(I_1) \subseteq f^*(I_1')$. Hence there exists $x' \in I_1'$ such that $f(x') = f(x)$. Thus by hypothesis on f we have $x \sqsubseteq x'$ and therefore, by definition of ideal, $x \in I_1'$.

Then we can conclude that f^* is also injective, thus it is a bijection, and clearly f^* as well as its inverse are monotone functions. $\square$

Notice that in particular, if P is a preorder, D is an algebraic CPO and $f : P \to \mathsf{K}(D)$ is a surjection such that $p \sqsubseteq p'$ iff $f(p) \sqsubseteq f(p')$, then $\mathsf{Idl}(P) \simeq \mathsf{Idl}(\mathsf{K}(D)) \simeq D$.

We can now prove the main result of this section, which establishes a tight relationship between the unfolding and the process semantics of semi-weighted c-nets. We show that the ideal completion of the preorder $(m \downarrow \mathbf{CP}[N])$ and the domain associated to the net N through the unfolding construction are isomorphic. To understand which is the meaning of taking the ideal completion of the preorder $(m \downarrow \mathbf{CP}[N])$, first notice that the elements of the partial order induced by the preorder $(m \downarrow \mathbf{CP}[N])$ are classes of concatenable processes with respect to an equivalence $\equiv_l$ defined by $\delta_1 \equiv_l \delta_2$ if there exist a discrete concatenable process δ such that $\delta_1;\delta = \delta_2$. In other words, $\delta_1 \equiv_l \delta_2$ can be read as "δ_1 and δ_2 left isomorphic", where "left" means that the isomorphism is required to be consistent only with respect to the ordering of the minimal places. Since the net N is semi-weighted, the equivalence $\equiv_l$ turns out to coincide with isomorphism of marked processes. In fact, being the initial marking of N a set, only one possible ordering function exists for the minimal places of a marked process. Finally, since processes are finite, taking the ideal completion of the partial order induced by the preorder $(m \downarrow \mathbf{CP}[N])$ (which produces the same result as taking directly the ideal completion of $(m \downarrow \mathbf{CP}[N])$) is necessary to move from finite computations to arbitrary ones.

Theorem 7 (unfolding vs. concatenable processes). *Let N be a semi-weighted c-net. Then the ideal completion of $(m \downarrow \mathbf{CP}[N])$ is isomorphic to the domain $\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N)))$.*

Proof. Let $N = \langle S, T, F, C, m\rangle$ be a c-net. It is worth recalling that the compact elements of the domain $\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N)))$, associated to N, are exactly the finite configurations of $\mathcal{E}_a(\mathcal{U}_a(N))$, as shown in Theorem 3. By Lemma 21, to prove the thesis it suffices to show that it is possible to define a function $\xi : (m \downarrow \mathbf{CP}[N]) \to \mathsf{K}(\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N))))$ such that f is *surjective*, and for all δ_1, δ_2 in $(m \downarrow \mathbf{CP}[N])$,

$$\delta_1 \lesssim \delta_2 \qquad \text{iff} \qquad \xi(\delta_1) \sqsubseteq \xi(\delta_2).$$

The function ξ can be defined as follows. Let $\delta = \langle \mu, \pi, \nu\rangle$ be a concatenable process in $(m \downarrow \mathbf{CP}[N])$. Being π a marked process of N (and thus a c-net morphism $\pi : O_\pi \to N$), by the universal property of coreflections, there exists a unique arrow $\pi' : O_\pi \to \mathcal{U}_a(N)$, making the diagram below commute.

$$\begin{tikzcd}
\mathcal{U}_a(N) \arrow[r, "f_N"] & N \\
O_\pi \arrow[u, dotted, "\pi'"] \arrow[ur, "\pi"'] &
\end{tikzcd}$$

In other words, the coreflection between **SW-CN** and **O-CN** gives a one-to-one correspondence between the (marked) processes of N and of those of its unfolding $\mathcal{U}_a(N)$.

Then we define $\xi(\delta) = \pi'_T(T_\pi)$, where T_π is the set of transitions of O_π. To see that ξ is a well defined function, just observe that it could have been written, more precisely, as $\mathcal{E}_a(\mathcal{U}_a(\pi))(T_\pi)$ and T_π is a configuration of $\mathcal{E}_a(\mathcal{U}_a(O_\pi)) = \mathcal{E}_a(O_\pi)$ since O_π is a deterministic occurrence c-net.

- ξ is *surjective*
  Let $C \in \mathsf{K}(\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N))))$ be a finite configuration. Then C determines a deterministic process $\pi'_C : O_{\pi'_C} \to \mathcal{U}_a(N)$ of the unfolding of N, having C as set of transitions.[15] Thus $\pi = f_N \circ \pi'_C$ is a deterministic process of N, and, by the definition of ξ, we immediately get that $\xi(\pi) = \pi'_C(T_{\pi'_C}) = C$.

- ξ is *monotone*
  Let δ_1 and δ_2 be processes in $(m \downarrow \mathbf{CP}[N])$ and let $\delta_1 \lesssim \delta_2$. Then, by Lemma 20 there exists a left-injection $\iota : \pi_1 \to \pi_2$. The picture below illustrates the situation, by depicting also the processes π'_1 and π'_2 of the unfolding of N, induced by π_1 and π_2, respectively.

$$\begin{tikzcd}
\mathcal{U}_a(N) \arrow[r, "f_N"] & N \\
O_{\pi_2} \arrow[u, dotted, "\pi'_2"] \arrow[ur, "\pi_2"'] & \\
O_{\pi_1} \arrow[u, "\iota"] \arrow[uur, "\pi_1"'] \arrow[uu, dotted, bend left, "\pi'_1"] &
\end{tikzcd}$$

  We have that $\xi(\delta_1) = \pi'_1(T_{\pi_1}) = \pi'_2(\iota(T_{\pi_1})) \subseteq \pi'_2(T_{\pi_2}) = \xi(\delta_2)$. Therefore, to conclude that $\xi(\delta_1) \sqsubseteq \xi(\delta_2)$ we must show that also the second condition of Definition 13 is satisfied. Let $t_2 \in \xi(\delta_2)$ and $t_1 \in \xi(\delta_1)$, with $t_2 \nearrow t_1$. By definition of ξ, $t_i = \pi_i(t'_i)$ with t'_i in O_{π_i}, for $i \in \{1, 2\}$ and thus:

[15] Essentially $O_{\pi'_C}$ is the obvious subnet of $\mathcal{U}_a(N)$ having C as set of transitions and π'_C is an inclusion.

$$\pi'_2(t'_2) \nearrow \pi'_1(t'_1) = \pi'_2(\iota(t'_1))$$

By properties of occurrence net morphisms (Corollary 3 and the fact that O_{π_2} is deterministic), this implies $t'_2 \nearrow \iota(t'_1)$ and thus, being ι a left injection, by rigidity $t'_2 = \iota(t)$ for some t in O_{π_1}. Therefore $t_2 = \pi'_2(t'_2) = \pi'_2(\iota(t)) = \pi'_1(t)$ belongs to $\xi(\delta_1)$, as desired.

– $\xi(\delta_1) \sqsubseteq \xi(\delta_2)$ implies $\delta_1 \lesssim \delta_2$.
Let $\xi(\delta_1) \sqsubseteq \xi(\delta_2)$. The inclusion $\xi(\delta_1) \subseteq \xi(\delta_2)$, immediately induces a mapping ι of the transitions of O_{π_1} into the transitions of O_{π_2}, defined by $\iota(t_1) = t_2$ if $\pi'_1(t_1) = \pi'_2(t_2)$ (see the picture above). This function is well-defined since processes are deterministic and thus morphisms π'_i are injective. Since the initial marking of N is a set, the mapping of $\min(\pi_1)$ into $\min(\pi_2)$ is uniquely determined and thus ι uniquely extends to a (marked) process morphism between π_1 and π_2 (more formally one should use Lemma 18). Again for the fact that N is semi-weighted (and thus there exists a unique indexing for the minimal places of each process starting from the initial marking) such morphism is consistent with the indexing of minimal places. Finally, ι is rigid. In fact, let $t_2 \nearrow \iota(t_1)$, for t_1 in O_{π_1} and t_2 in O_{π_2}. By properties of occurrence c-net morphisms (Lemma 14), $\pi'_2(t_2) \nearrow \pi'_2(\iota(t_1))$. The way ι is defined implies that $\pi'_2(\iota(t_1)) = \pi'_1(t_1)$, and thus

$$\pi'_2(t_2) \nearrow \pi'_1(t_1).$$

Since $\pi'_i(t_i) \in \xi(\delta_i)$ for $i \in \{1,2\}$, by definition of the order on configurations, we immediately have that $\pi'_2(t_2) \in \xi(\delta_1)$, thus there is t'_1 in O_{π_1} such that $\pi'_1(t'_1) = \pi'_2(t_2)$, and thus $\iota(t'_1) = t_2$.
By lemma 20, the existence of the left injection $\iota : \delta_1 \to \delta_2$, implies $\delta_1 \lesssim \delta_2$.

□

9 Conclusions and future work

The main contribution of this paper is a truly concurrent event-based semantics for (semi-weighted) P/T contextual nets. The semantics is given at categorical level via a coreflection between the categories **SW-CN** of semi-weighted c-nets and **Dom** of finitary coherent prime algebraic domains (or equivalently **PES** of prime event structures). Such a coreflection factorizes through the following chain of coreflections:

$$\begin{tikzcd}
\mathbf{SW\text{-}CN} \arrow[r, shift right=1.5, "\mathcal{U}_a"'] \arrow[r, phantom, "\perp"] & \mathbf{O\text{-}CN} \arrow[l, hook', shift right=1.5, "\mathcal{I}_O"'] \arrow[r, shift right=1.5, "\mathcal{E}_a"'] \arrow[r, phantom, "\perp"] & \mathbf{AES} \arrow[l, shift right=1.5, "\mathcal{N}_a"'] \arrow[r, shift right=1.5, "\mathcal{L}_a"'] \arrow[r, phantom, "\perp"] & \mathbf{Dom} \arrow[l, shift right=1.5, "\mathcal{P}_a"'] \arrow[r, shift right=1.5, "\mathcal{P}"'] \arrow[r, phantom, "\sim"] & \mathbf{PES} \arrow[l, shift right=1.5, "\mathcal{L}"']
\end{tikzcd}$$

It is worth noticing that such a construction is a consistent extension of Winskel's one [39], in the sense that it associates to a safe c-net without context places the same occurrence net and domain produced by Winskel's construction.

We also showed that a close relationship exists between the unfolding semantics and the deterministic process semantics, generalizing a result of [25] to c-nets. Roughly speaking, the domain associated to a semi-weighted contextual net by the above functors is shown to be isomorphic to the set of deterministic processes of the net starting from the initial marking, endowed with a kind of prefix ordering.

A key rôle in our semantics is played by asymmetric event structures, an extension of Winskel's (prime) event structures (with binary conflict), introduced to deal with asymmetric conflicts. Asymmetric event structures are closely related to other models in the literature, like PES's with possible events [32], flow event structures with possible flow [32] and extended bundle event structures [23]. However, none of the above models was adequate for our aims: PES's with possible events are not sufficiently expressive, while the other two models look too general and unnecessarily complex for the concerns of this paper, due to their capability of expressing multiple disjunctive causes for an event. Moreover, no categorical treatment of the more general models was available and, due to their greater complexity, it is still unclear if the coreflection result between **AES** and **Dom** of this paper extends to them. Understanding which part of the results presented in this paper for AES's extends to flow event structures with possible flow and to bundle event structures with asymmetric conflict is an interesting matter of further investigation.

The unfolding construction associating an occurrence c-net to a semi-weighted c-net, presented in this paper, is a proper extension of the construction developed in an independent work by Vogler et al. [37]. Apart from the fact that the unfolding of [37] is carried out in the case of safe finite c-nets, the major difference is the fact that our unfolding is defined at categorical level and establishes a coreflection between the categories of semi-weighted and occurrence c-nets. This result ensures the naturality of the semantics [18] (the unfolding is the unique right adjoint to the inclusion functor) as well as its compositionality with respect to operations on nets defined in terms of limits (e.g., synchronization).

We already mentioned that Winskel's construction has been generalized in [25] not only to the subclass of semi-weighted P/T nets, but also to the full class of P/T nets. In the last case, some additional effort is needed and only a proper adjunction rather than a coreflection can be obtained. We are quite confident that also the results of this paper could be extended to the full class of P/T contextual nets, by following the guidelines traced in [25] and exploiting, in particular, a suitable generalization to c-nets of the notions of decorated occurrence net and family morphism introduced in that work.

Apart from the application to c-nets analyzed in this paper, asymmetric event structures seem to be rather promising in the semantic treatment of models of computation, such as string, term and graph rewriting, allowing context sensitive firing of events. Therefore, as suggested in [32], it would be interesting to investigate the possibility of developing a theory of general event structures with asymmetric conflict (or weak causality) similar to that in [39].

Finally, we remark that one of the motivations of the research on contextual nets is their relationship with *graph transformation systems* (GTS's) [16, 13], a formalism for the specification of concurrent and distributed systems which can be an appropriate alternative to Petri nets when one is interested in having a more structured description of the state. In fact, in a GTS the state is represented by a graph and local transformations of the state are modelled via the application of graph productions, which, roughly speaking, are rules specifying that the left-hand side of the rule, in a given context, rewrites to its right-hand side. Since Petri nets are essentially rewriting systems on multisets, it is quite natural to see GTS's as a proper extension of Petri nets both for the fact that they allow for a more complex state and for their capability of expressing "contextual" rewritings. It is worth noting that, in the case of GTS's, "contexts" are not an optional feature but an essential part of the rewriting mechanism, which permits to specify how the subgraph added by the step is connected to the remaining part of the state. To better understand this fact, recall that, according to [16], a graph production consists of a left-hand side graph L, a right-hand side graph R and a (common) interface graph K embedded both in R and in L, as depicted in the top part of Figure 10. Informally, to apply such a rule to a graph G we must find an occurrence of its

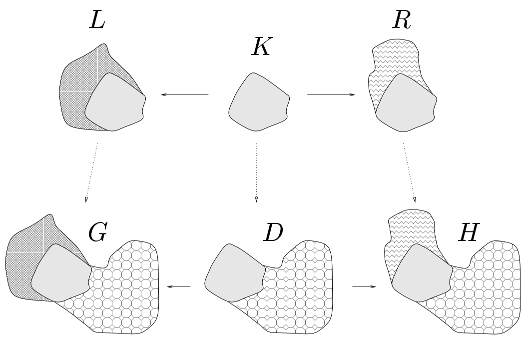


Fig. 10. A (double pushout) graph rewriting step.

left-hand side L in G. The rewriting mechanism first removes the part of the left-hand side L which is not in the interface K producing the graph D, and then adds the part of the right-hand side R which is not in the interface K, thus obtaining the graph H. The interface graph K is "preserved": it is necessary to perform the rewriting step, but it is not affected by the step itself, and as such it corresponds to the contexts of our contextual nets. Notice that the interface K plays a fundamental rôle in specifying how the right-hand side has to be glued with the graph D. Working without contexts, which in a grammar-theoretical framework would mean working with productions having an empty interface graph K, the expressive power of graph grammars would drastically decrease: only disconnected subgraphs could be added.

To present GTS's as a formalism for concurrent/distributed systems, people working in this area have been naturally lead to the attempt of providing them with an appropriate concurrent semantics. In particular, some efforts have been spent in the direction of recasting in this more general framework notions, constructions and results from Petri nets theory. Unfortunately, the reason for which graph grammars represent an appealing generalization of Petri nets, namely the fact that they extend nets with some non-trivial features, makes non-trivial also such generalization. Some successful results in the project of extending the constructions from net theory to GTS's have been obtained in the developement of a theory of non sequential processes for GTS's [12, 2]. Since contextual nets extend traditional nets with one of the new features of GTS's, namely with the capability of preserving part of the state in a rewriting step, we think that the work on c-nets could help in transferring notions and results from nets to GTS. Indeed, (part of) the results of this paper have been recasted for GTS's, but only at "object level" [1]. A categorical treatment is still missing and constitutes a direction of further research.

References

1. P. Baldan, A. Corradini, and U. Montanari. Unfolding and event structure semantics for graph grammars. To appear in the *Proceedings of FoSSaCS'99.*
2. P. Baldan, A. Corradini, and U. Montanari. Concatenable graph processes: relating processes and derivation traces. In *Proceedings of ICALP'98*, volume 1443 of *LNCS*, pages 283–295. Springer Verlag, 1998.
3. P. Baldan, A. Corradini, and U. Montanari. An event structure semantics for P/T contextual nets: Asymmetric event structures. In M. Nivat, editor, *Proceedings of FoSSaCS '98*, volume 1378, pages 63–80. Springer Verlag, 1998.
4. G. Berry. Stable models of typed lambda-calculi. In *Proceeding of the 5^{th} ICALP*, volume 62 of *LNCS*, pages 72–89. Springer-Verlag, 1978.
5. G. Boudol. Flow Event Structures and Flow Nets. In *Semantics of System of Concurrent Processes*, volume 469 of *LNCS*, pages 62–95. Springer Verlag, 1990.
6. G. Boudol and I. Castellani. Permutation of transitions: an event structure semantics for CCS and SCCS. In *Linear Time, Branching Time and Partial Order Semantics in Logics and Models for Concurrency*, volume 354 of *LNCS*, pages 411–427. Springer Verlag, 1988.
7. N. Busi. *Petri Nets with Inhibitor and Read Arcs: Semantics, Analysis and Application to Process Calculi.* PhD thesis, University of Siena, Department of Computer Science, 1998.
8. N. Busi and R. Gorrieri. A Petri Nets Semantics for π-calculus. In *Proceedings CONCUR'95*, volume 962 of *LNCS*, pages 145–159. Springer Verlag, 1995.
9. N. Busi and G. M. Pinna. Non Sequential Semantics for Contextual P/T Nets. In *Application and Theory of Petri Nets*, volume 1091 of *LNCS*, pages 113–132. Springer Verlag, 1996.
10. S. Christensen and N. D. Hansen. Coloured Petri nets extended with place capacities, test arcs and inhibitor arcs. In M. Ajmone-Marsan, editor, *Applications and Theory of Petri Nets*, volume 691 of *LNCS*, pages 186–205. Springer Verlag, 1993.
11. A. Corradini. Concurrent Graph and Term Graph Rewriting. In U. Montanari and V. Sassone, editors, *Proceedings CONCUR'96*, volume 1119 of *LNCS*, pages 438–464. Springer Verlag, 1996.
12. A. Corradini, U. Montanari, and F. Rossi. Graph processes. *Fundamenta Informaticae*, 26:241–265, 1996.
13. A. Corradini, U. Montanari, F. Rossi, H. Ehrig, R. Heckel, and M. Löwe. Algebraic Approaches to Graph Transformation I: Basic Concepts and Double Pushout Approach. In G. Rozenberg, editor, *Handbook of Graph Grammars and Computing by Graph Transformation. Volume 1: Foundations.* World Scientific, 1997.
14. P. Degano, R. Gorrieri, and S. Vigna. On Relating Some Models for Concurrency. In M. C. Gaudel and J. P. Jouannaud, editors, *4^{th} Conference on Theory and Practice of Software Development*, volume 668 of *LNCS*, pages 15–30. Springer-Verlag, 1993.
15. P. Degano, J. Meseguer, and U. Montanari. Axiomatizing Net Computations and Processes. In *Proc. 4th Annual Symposium on Logic in Computer Science, Asilomar, CA, USA*, pages 175–185, 1989.
16. H. Ehrig. Tutorial introduction to the algebraic approach of graph-grammars. In H. Ehrig, M. Nagl, G. Rozenberg, and A. Rosenfeld, editors, *Proceedings of the 3rd International Workshop on Graph-Grammars and Their Application to Computer Science*, volume 291 of *LNCS*, pages 3–14. Springer Verlag, 1987.
17. F. Gadducci and U. Montanari. Axioms for contextual net processes. In *Proceedings ICALP'98*, LNCS, pages 296–308. Springer Verlag, 1998.
18. J.A. Goguen. A categorical manifesto. *Mathematical Structures in Computer Science*, 1, 1991.

19. R. Janicki and M. Koutny. Invariant semantics of nets with inhibitor arcs. In *Proceedings CONCUR '91*, volume 527 of *LNCS*. Springer Verlag, 1991.
20. R. Janicki and M. Koutny. Structure of concurrency. *Theoret. Comput. Sci.*, 112:5–52, 1993.
21. R. Janicki and M. Koutny. Semantics of inhibitor nets. *Information and Computation*, 123:1–16, 1995.
22. R. Langerak. Bundle Event Structures: A Non-Interleaving Semantics for Lotos. In 5^{th} *Intl. Conf. on Formal Description Techniques (FORTE'92)*, pages 331–346. North-Holland, 1992.
23. R. Langerak. *Transformation and Semantics for LOTOS*. PhD thesis, Department of Computer Science, University of Twente, 1992.
24. J. Meseguer, U. Montanari, and V. Sassone. On the semantics of Petri nets. In *Proceedings CONCUR '92*, volume 630 of *LNCS*, pages 286–301. Springer Verlag, 1992.
25. J. Meseguer, U. Montanari, and V. Sassone. Process versus unfolding semantics for Place/Transition Petri nets. *Theoretical Computer Science*, 153(1-2):171–210, 1996.
26. J. Meseguer, U. Montanari, and V. Sassone. On the semantics of Place/Transition Petri nets. *Mathematical Structures in Computer Science*, 7:359–397, 1997.
27. U. Montanari and F. Rossi. Contextual occurrence nets and concurrent constraint programming. In H.-J. Schneider and H. Ehrig, editors, *Proceedings of the Dagstuhl Seminar 9301 on Graph Transformations in Computer Science*, volume 776 of *LNCS*. Springer Verlag, 1994.
28. U. Montanari and F. Rossi. Contextual nets. *Acta Informatica*, 32, 1995.
29. M. Nielsen, G. Plotkin, and G. Winskel. Petri Nets, Event Structures and Domains, Part 1. *Theoret. Comput. Sci.*, 13:85–108, 1981.
30. C.A. Petri. *Kommunikation mit Automaten*. Schriften des Institutes für Instrumentelle Matematik, Bonn. 1962.
31. G. M. Pinna and A. Poigné. On the nature of events. In *Mathematical Foundations of Computer Science*, volume 629 of *LNCS*, pages 430–441. Springer Verlag, 1992.
32. G. M. Pinna and A. Poigné. On the nature of events: another perspective in concurrency. *Theoretical Computer Science*, 138:425–454, 1995.
33. W. Reisig. *Petri Nets: An Introduction*. EACTS Monographs on Theoretical Computer Science. Springer Verlag, 1985.
34. G. Ristori. *Modelling Systems with Shared Resources via Petri Nets*. PhD thesis, Università di Pisa, 1994.
35. W. Vogler. Efficiency of asynchronous systems and read arcs in Petri nets. Technical Report 352, Institüt für Mathematik, Augsburg University, 1996.
36. W. Vogler. Partial Order Semantics and Read Arcs. In *Mathematical Foundations of Computer Science*, volume 1295 of *LNCS*, pages 508–518. Springer Verlag, 1997.
37. W. Vogler, A. Semenov, and A. Yakovlev. Unfolding and finite prefix for nets with read arcs. In *Proceedings of CONCUR'98*, volume 1466 of *LNCS*, pages 501–516. Springer-Verlag, 1998.
38. J. Winkowsky. Processes of Contextual Nets and their Characteristics. Technical Report 847, Institute of Computer Science, Polish Academy of Sciences, 1998.
39. G. Winskel. Event Structures. In *Petri Nets: Applications and Relationships to Other Models of Concurrency*, volume 255 of *LNCS*, pages 325–392. Springer Verlag, 1987.