Information and Computation **171**, 1–49 (2001)
doi:10.1006/inco.2001.3060, available online at http://www.idealibrary.com on IDEAL®

Contextual Petri Nets, Asymmetric Event Structures, and Processes[1]

Paolo Baldan, Andrea Corradini, and Ugo Montanari

Dipartimento di Informatica, University of Pisa Corso Italia, 40, 56125 Pisa, Italy
E-mail: baldan@di.unipi.it, andrea@di.unipi.it, ugo@di.unipi.it

Received February 11, 1999

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 role 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. © 2001 Elsevier Science

Key Words: contextual Petri nets; read arcs; asymmetric conflict; concurrent semantics; unfolding; event structures; domains; processes.

1. INTRODUCTION

Petri nets are widely accepted as an adequate formalism for the specification of the behaviour of concurrent and distributed systems [25, 41]. 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 its preconditions, and, in this case, the firing of the transition *removes* such tokens and *produces* new tokens in its postconditions. More transitions can fire together when they consume mutually disjoint sets of tokens. This informal description should already suggest how Petri nets can specify in a natural way phenomena such as mutual exclusion, concurrency, sequential composition, and nondeterminism.

A limit in the expressiveness of Petri nets is represented by the fact that transitions can only *consume* and *produce* tokens, and thus a net cannot express in a natural way nondestructive 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 Fig. 1, where place s is intended to represent a resource which is accessed by two transitions t_0 and t_1 in a read-only modality. Different from what one could expect the two transitions cannot read the instance of the shared resource s concurrently, but their accesses must be serialized.

Contextual nets. *Contextual nets* [20, 23], also called nets with test arcs [18], with activator arcs [11], or with read arcs [27], 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 net has also some *context* conditions that, informally speaking, specify that the transition to be enabled requires the presence of some tokens, which, however, 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

[1] Research partly supported by the EC TMR Network *General Theory of Graph Transformation* (GETGRATS), by the Esprit Working Group *Applications of Graph Transformation* (APPLIGRAPH), and by the MURST Project *Tipi, Ordine Superiore e Concorrenza* (TOSCA).

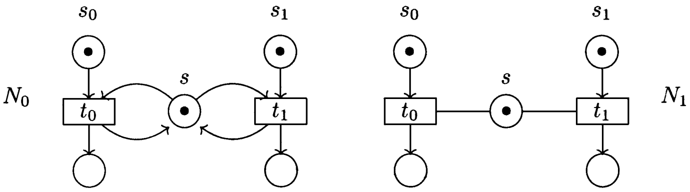


FIG. 1. Ordinary nets do not allow for concurrent read-only operations.

postconditions being simply written. Coherently with this view, the same token can be used as context by many transitions at the same time. For instance, the situation of two agents reading a shared resource discussed above can be modelled directly by the contextual net N_1 of Fig. 1, where the transitions t_0 and t_1 use the place s as context. According to the informal description of the behaviour 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 [20], nondirected (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 accesses to shared data (e.g., reading in a database) [26], to provide concurrent semantics to concurrent constraint (CC) programs [16], to model priorities [24], and to specify a net semantics for the π-calculus [36]. Moreover, they have been studied for their connections with another powerful formalism for the specification of concurrent computations, namely graph transformation systems [17, 20]. 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 notions and results developed for nets (see, e.g., [12, 32, 42]).

In his seminal work [10], Winskel, starting from some results in [1], shows that an event structure semantics for *safe* nets can be given via a chain of coreflections leading from the category **Safe** of safe nets to the category **PES** of prime event structures, 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 causality and conflict relations among them. In [14, 31] 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 postcondition. 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 two examples of semi-weighted P/T nets which are not safe. Interestingly, from the point of view of expressiveness, semi-weighted nets allow one to model an unbounded degree of concurrency, which instead is not expressible in safe nets. For instance, in the semi-weighted net N_2' of Fig. 2, after n firings of transition t_0, the place s contains n tokens and thus n copies of t_1 can fire in parallel.

This paper generalizes such results to the setting of *contextual nets* by showing that an event structure for a semi-weighted contextual net,[2] describing its concurrent behaviour, 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 arise in such a development and the way we have decided to solve them.

Asymmetric conflicts and asymmetric event structures. *Prime event structures* (PES's) 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

[2] Semi-weighted nets were called "weakly-safe nets" in the conference version of this paper [19].

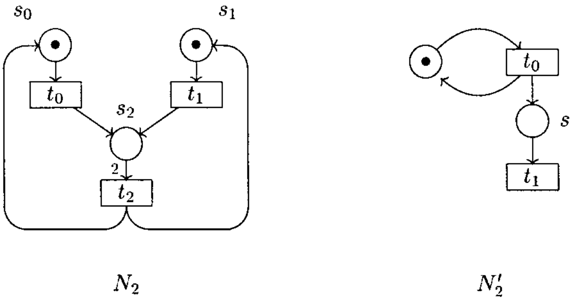


FIG. 2. Two semi-weighted P/T nets, which are not safe.

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 Fig. 3a, with two transitions t_0 and t_1 which use the same place s as context and precondition, respectively.

The possible firing sequences are given by the firing of t_0, the firing of t_1, and the firing of t_0 followed by t_1, denoted $t_0; t_1$, while $t_1; t_0$ is not allowed. Also the concurrent firing of t_0 and t_1 is not possible, different from what happens in [11] and [27], the idea being that two concurrent events should be allowed to occur also in any order. This situation cannot be modelled in a direct way within a prime event structure: t_0 and t_1 are neither in conflict nor concurrent nor causally dependent. Simply, as for an ordinary conflict, the firing of t_1 prevents t_0 being 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 fire, in such computations t_0 acts as a cause of t_1. However, different 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 *causality*.

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 Fig. 3b, where causality 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 single event. The proposed 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 the information on the new kind of dependency determined by read-only operations is completely lost, because it 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 [10, Definition 1.1.1]) stating that the enabling relation $\vdash$ is

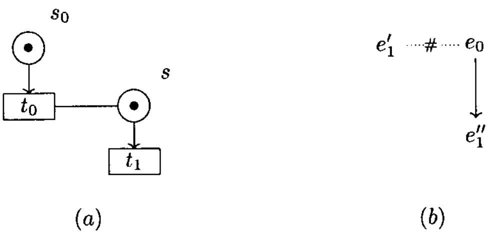


FIG. 3. A simple contextual net and a prime event structure representing its behaviour.

"monotone" with respect to set inclusion:

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

As a consequence, the computational order between configurations is set inclusion, the idea being that if A and B are finite configurations such that $A \subseteq B$, 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 net computations we introduce a new kind of event structure, called *asymmetric event structure* (AES). An AES, besides the usual causality relation $\leq$ of a prime event structure, has a relation $\nearrow$ that allows us to specify the new kind of dependency described above. E.g., for the transitions t_0 and t_1 of the net in Fig. 3 we simply have $t_0 \nearrow t_1$. As already noted, 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 have 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 causality 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, different from PES's, an event of an AES 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 cannot be extended with an event inhibited by other events already present in C. Such a construction extends to a functor from the category **AES** of asymmetric event structures to the category **Dom** of prime algebraic domains that establishes a coreflection between **AES** and **Dom**. By using the equivalence between the category **Dom** and the category **PES** of prime event structures [10] we can then translate any AES into an ordinary PES. Essentially the PES obtained in this way encodes the asymmetric conflict by means of causality and symmetric conflict, as depicted in Fig. 3. Observe that the AES provides a finer semantics than the PES, since different AES's may be mapped to the same PES. It is remarkable that the "translation" from AES's to PES's is done at a 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 nondeterministic choice or, equivalently, the possibility of having multiple disjunctive and mutually exclusive causes for an event, Boudol and Castellani [15] introduce the notion of *flow event structure*, where the causality relation is replaced by an irreflexive (in general nontransitive) *flow relation*, representing essentially immediate causal dependency, and conflict is no longer hereditary. To face a similar problem, Langerak [7] 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 [2, 43], 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 *et al.* in [21], where prioritized event structures are introduced as PES's enriched with a partial order relation modeling priorities between events. Also bundle event structures have been extended by Langerak in [33] to take into account asymmetric conflicts.

Despite some differences in the definition and in the related notions, our AES's 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 ordinary 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 the 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 $\mathcal{I}_{oc}$: **O-CN** → **SW-CN**.

Transitions of an occurrence contextual net are related by causality 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 by simply forgetting the places, but remembering the dependency relations that they induce between transitions. Again, this construction extends, at a 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's [10] and flow nets represent flow event structures [40]. Finally, the coreflection between **AES** and **Dom**, discussed above, can be exploited to complete the chain of coreflections from **SW-CN** to **Dom**.

Independent from the conference version of this paper, which appeared as [19], an unfolding construction for (safe finite) contextual nets has been proposed by Vogler *et al.* in [5]. Apart from some matters of presentation, the construction in [5] is based on ideas analogous to ours and it leads, for the considered class of nets, to the same unfolding. An interesting result in the mentioned paper, witnessing the practical relevance of the study of the semantics of contextual nets, is the generalization to a subclass of safe contextual nets, called read-persistent contextual nets, of McMillan's algorithm [35] for the construction of a (complete) finite prefix of the unfolding. The algorithm is then applied to the analysis of asynchronous logic circuits, showing that the use of contexts allows one to model a circuit via a simpler net with a smaller unfolding, thus making the verification activity more efficient.

The study of the applications of the concurrent semantics of contextual nets goes beyond the goals of the present paper. Concerning the unfolding construction, the main differences between [5] and our approach are that we deal with a slightly larger class of nets (including possibly infinite semi-weighted nets) and that we provide a categorical characterization of the unfolding as a coreflection. We think that the advantages of having a categorical semantics defined via an adjunction are numerous. First, one is led 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 a 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 a 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 nondeterministic composition, while synchronization of nets can be modeled as a product (see [10, 14]). Since left–right adjoint functors preserve colimits–limits, a semantics defined via an adjunction turns out to be compositional with respect to such operations. An interesting discussion on the usefulness of category theory in computer science can be found in Goguen's paper [13].

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., [6, 8, 20, 24, 29, 39]). 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 ordinary nets [9], 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 [29, 39]. 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. Interestingly, 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 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 shows how the proposed semantics for semi-weighted contextual nets is related to Winskel's semantics for safe ordinary nets and comments on the expressive power of semi-weighted and safe contextual nets. Section 9 investigates the relation between the unfolding and the deterministic process semantics of contextual nets. Section 10 discusses how the results presented in this paper can be extended to deal with a wider class of contextual nets, where contexts might have multiplicities. Finally, Section 11 draws some conclusions and suggests possible directions for further research. An extended abstract of Sections 2–7 appeared in [19].

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. 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. 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 (nontrivial) 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 mean 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. When $Y \in \mathbf{2}^X_{fin}$ we will write $Y \subseteq_{fin} X$.

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

Definition 2.1 (Prime Event Structure). 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 $\lfloor e \rfloor = \{e' \in E \mid e' \leq e\}$ is finite for all $e \in E$;
2. the relation $\#$ is irreflexive, symmetric, and hereditary with respect to $\leq$, i.e., for all $e, e', e'' \in E$, if $e \# e' \leq e''$ then $e \# e''$;

Let $P_0 = \langle E_0, \leq_0, \#_0 \rangle$ and $P_1 = \langle E_1, \leq_1, \#_1 \rangle$ be PES's. A PES-morphism $f : P_0 \to P_1$ is a partial function $f : E_0 \to E_1$ such that, for all $e_0, e_0' \in E_0$:

1. if $f(e_0) \neq \bot$ then $\lfloor f(e_0) \rfloor \subseteq f(\lfloor e_0 \rfloor)$;
2. if $f(e_0) \neq \bot \neq f(e_0')$ then
   (i) $f(e_0) \#_1 f(e_0') \Rightarrow e_0 \#_0 e_0'$;
   (ii) $(f(e_0) = f(e_0')) \wedge (e_0 \neq e_0') \Rightarrow e_0 \#_0 e_0'$.

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 [2, 43]. In these 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 one to specify adequately features such as priority and asymmetric conflict, they introduce the idea of possible events, namely events that, according to the considered computation, may or may not be causes of other events. Consequently the notions of PES with possible events and 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's 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, different 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 [33] 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.

For technical reasons we first introduce pre-asymmetric event structures. Then asymmetric event structures will be defined as special pre-asymmetric event structures satisfying a suitable condition of "saturation."

Definition 2.2 (Pre-asymmetric Event Structure). A *pre-asymmetric event structure (pre-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 $\lfloor e \rfloor = \{e' \in E \mid e' \leq e\}$ is finite for all $e \in E$;
2. the relation $\nearrow$ satisfies, for all $e, e' \in E$,
   (i) $e < e' \Rightarrow e \nearrow e'$,
   (ii) $\nearrow_{\lfloor e \rfloor}$ is acyclic,[3]

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

where, as usual, with $e < e'$ we mean $e \leq e'$ and $e \neq e'$. 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', written $e \leadsto_G e'$, if $e \nearrow e'$ and $\neg(e < e')$.

The definition can be explained by giving a more precise account of the ideas presented in the introduction. Let $occur(e, C)$ denote the fact that the event e occurs in a computation C, later formalized by the notion of configuration, and let $prec_C(e, e')$ indicate that the event e precedes e' in C. Then, informally,

$$\begin{array}{lll} e < e' & \text{means that} & \forall C.\ occur(e', C) \Rightarrow occur(e, C) \wedge prec_C(e, e') \\ e \nearrow e' & \text{means that} & \forall C.\ occur(e', C) \wedge occur(e, C) \Rightarrow prec_C(e, e'). \end{array}$$

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 some events form a cycle of asymmetric conflict 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 also makes 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 e' also fires.

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

Definition 2.3 (Induced Conflict Relation). Let $G = \langle E, \leq, \nearrow \rangle$ be a pre-AES. The *conflict relation* $\#^a \subseteq \mathbf{2}^E_{fin}$ associated to G is defined as

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

where A is a finite subset of E. The superscript "a" in $\#^a$ reminds us that this relation is induced by asymmetric conflict. Sometimes we will 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, vice versa, 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 $i \in \{0, \ldots, n\}$, the set A' can be $\{a_i \mid i \in \{0, \ldots, n\}\}$).

Clearly, by the rules above, if $e \nearrow e'$ and $e' \nearrow e$ then $\#^a\{e, e'\}$. The converse, instead, does not hold, namely in general we can have $e \#^a e'$ and $\neg(e \nearrow e')$, as in the AES of Fig. 4, because $\#^a$ is inherited along $\leq$, while $\nearrow$ is not. An asymmetric event structure is a pre-AES where each binary conflict is induced directly by an asymmetric conflict in both directions.

Definition 2.4 (Asymmetric Event Structures). An asymmetric event structure (AES) is a pre-AES $G = \langle E, \leq, \nearrow \rangle$ such that for any $e, e' \in E$, if $e \#^a e'$ then $e \nearrow e'$.

Observe that any pre-AES can be saturated to produce an AES. More precisely, given a pre-AES $G = \langle E, \leq, \nearrow \rangle$, its saturation, denoted by $\overline{G}$, is the AES $\langle E, \leq, \nearrow' \rangle$, where $\nearrow'$ is defined as $e \nearrow' e'$ if and only if $(e \nearrow e') \vee (e \#^a e')$. In this situation it is easy to verify that the conflict relations of G and of $\overline{G}$ coincide.

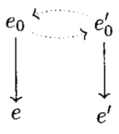


FIG. 4. A pre-AES with two events e and e' in conflict, but not related by asymmetric conflict.

The notion of AES-morphism is a quite natural extension of the notion of PES-morphism. Intuitively, it is a (possibly partial) mapping of events that "preserves computations," a property which will be made precise later, in Lemma 3.6, after introducing the notion of configuration.

DEFINITION 2.5 (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, for all $e_0, e_0' \in E_0$:

1. if $f(e_0) \neq \bot$ then $\lfloor f(e_0) \rfloor \subseteq f(\lfloor e_0 \rfloor)$;
2. if $f(e_0) \neq \bot \neq f(e_0')$ then
   (i) $f(e_0) \nearrow_1 f(e_0') \Rightarrow e_0 \nearrow_0 e_0'$;
   (ii) $(f(e_0) = f(e_0')) \wedge (e_0 \neq e_0') \Rightarrow e_0 \#_0^a e_0'$.

It is easy to show that AES-morphisms are closed under composition. In fact, let $f_0 : G_0 \to G_1$ and $f_1 : G_1 \to G_2$ be AES-morphisms. The fact that $f_1 \circ f_0$ satisfies conditions (1) and (2.ii) of Definition 2.5 is proved as for ordinary PES's. The validity of condition (2.i) is straightforward.

DEFINITION 2.6 (Category **AES**). We denote by **AES** the category having asymmetric event structures as objects and AES-morphisms as arrows.

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 and relations (e.g., $G_i = \langle E_i, \leq_i, \nearrow_i \rangle$).

The binary conflict in an AES is represented by asymmetric conflict in both directions, and thus, analogously to what happens for PES's, it is reflected by AES-morphisms (by condition (2.i) in Definition 2.5). The next lemma shows that AES-morphisms reflect also the general conflict relation over sets of events.

LEMMA 2.1 (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 $\#_1^a f(A)$. By definition of conflict there is a $\nearrow_1$-cycle $e_0' \nearrow_1 e_1' \nearrow_1 \ldots \nearrow_1 e_n' \nearrow_1 e_0'$ in $\lfloor f(A) \rfloor$. By the definition of AES-morphisms we have that $\lfloor f(A) \rfloor \subseteq f(\lfloor A \rfloor)$ and thus we can find $e_0, \ldots, e_n \in \lfloor A \rfloor$ such that $e_i' = f(e_i)$ for all $i \in \{0, \ldots, n\}$. Consider $A' = \{a_0, \ldots, a_n\} \subseteq A$ such that $e_i \leq_0 a_i$ for $i \in \{0, \ldots, n\}$. By definition of AES-morphism, $e_0 \nearrow_0 e_1 \nearrow_0 \ldots \nearrow_0 e_0$, and thus $\#_0^a A'$. ■

We conclude this section by formalizing the relation between AES's and PES's. We show that AES's are a proper extension of PES's in the sense that, as one would expect, PES's can be identified with the subclass of AES's where the strict asymmetric conflict relation is actually symmetric. This correspondence defines a full embedding of **PES** into **AES**.

LEMMA 2.2. *Let $P = \langle E, \leq, \# \rangle$ be a* PES*. Then $\mathcal{J}(P) = \langle E, \leq, < \cup \# \rangle$ 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 $\mathcal{J}(P_0)$ and $\mathcal{J}(P_1)$, and if $g : \mathcal{J}(P_0) \to \mathcal{J}(P_1)$ 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 $\mathcal{J}(P) = \langle E, \leq, < \cup \# \rangle$ is an AES is a trivial consequence of the definitions. In particular, the asymmetric conflict relation of $\mathcal{J}(P)$ is acyclic on the causes of each event since # is hereditary with respect to $\leq$ and irreflexive, and $<$ is a strict partial order (i.e., an irreflexive and transitive relation) 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's $\mathcal{J}(P_0)$ and $\mathcal{J}(P_1)$, first observe that, according to the definition of $\leq_{\mathcal{J}(Pi)}$ and $\nearrow_{\mathcal{J}(P_i)}$, the validity of the conditions (1) and (2.ii) of Definition 2.5 follow immediately from the corresponding conditions in the definition of PES-morphism (Definition 2.1). As for Condition (2.i), if $f(e_0) \nearrow_{\mathcal{J}(P_1)} f(e_1)$, then, by construction, $f(e_0) <_{P_1} f(e_1)$ or $f(e_0) \#_{P_1} f(e_1)$ and thus, by properties of PES's (easily derivable from Definition 2.1), in the first case $e_0 <_{P_0} e_1$ or $e_0 \#_{P_0} e_1$ whilst, in the second case, $e_0 \#_{P_0} e_1$. Hence, in both cases, $e_0 \nearrow_{\mathcal{J}(P_0)} e_1$.

Similar considerations allow us to conclude that if $g : \mathcal{J}(P_0) \to \mathcal{J}(P_1)$ is an AES-morphism, then it is also a PES-morphism between P_0 and P_1. ■

By the previous lemma, the construction $\mathcal{J}$, extended as the identity on arrows, defines a full embedding functor from **PES** into **AES**.

PROPOSITION 2.1 (From PES's to AES's). *The functor* $\mathcal{J}$: **PES** → **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 Fig. 3).

3.1. Prime Event Structures and Domains

This section reviews the definition of the category **Dom** and the equivalence between **Dom** and the category **PES** [10], which will be needed in the remainder of the paper.

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 3.7 ((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 3.8 ((Prime Algebraic) Coherent Poset). A partial order D is called *coherent* (or *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 $Pr(d)$.

Not being 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 nonempty $X \subseteq D$ 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 3.9 (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 a relation.

Definition 3.10 (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.

In the paper [10] the category **Dom** is shown to be equivalent to the category **PES**, the equivalence being established by the two functors $\mathcal{L}$: **PES** → **Dom** and $\mathcal{P}$: **Dom** → **PES**

$$\begin{tikzcd} \mathbf{PES} \arrow[r, shift right, "\mathcal{L}"'] \arrow[r, phantom, "\sim"] & \mathbf{Dom}. \arrow[l, shift right, "\mathcal{P}"'] \end{tikzcd}$$

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 in the next section to map domains back into asymmetric event structures. A fundamental role is played by the notion of prime interval.

Definition 3.11 (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

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

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 [1]). More precisely, the map

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

[4] The use of this kind of structure in semantics was first investigated by Berry [28], 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.

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 3.12 (From Domains to PES's). The functor $\mathcal{P}$: **Dom** → **PES** is defined as follows:

- given a domain D, $\mathcal{P}(D) = \langle Pr(D), \leq, \# \rangle$ where
$$p \leq p' \quad \text{iff} \quad p \sqsubseteq p' \qquad \text{and} \qquad p \# p' \quad \text{iff} \quad \neg(p \uparrow p');$$
- 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:

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

3.2. Asymmetric Event Structures and Domains

This section defines a coreflection between the category **AES** and the category **Dom**. The domain associated to an AES G is obtained by considering the configurations of G, suitably ordered using the asymmetric conflict relation. Vice versa, given a domain D we obtain the corresponding AES by applying first the functor $\mathcal{P}$: **Dom** → **PES** and then the embedding $\mathcal{J}$: **PES** → **AES**, defined in Proposition 2.1.

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

Definition 3.13 (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 (3) requires that all the causes of each event are present. Condition (1) first ensures that in C there are no $\nearrow$-cycles, and thus, together with (3), it 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), imply 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.

If a set of events A satisfies only the first two properties of Definition 3.13 it is called *consistent* and we write $co(A)$. Notice that, unlike for Winskel's 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 Winskel's event structures 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.

Definition 3.14 (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

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

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

Often in the following 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 [10], flow event structure [40], and prioritized event structures [21], asymmetric event structures also 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 3.1. *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 $e \in C$*

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

Proof. Let $e' \in \bigcup A$ be an event such that $e' \nearrow e$. Then there is a configuration $C' \in A$ such that $e' \in C'$. Since C and C' are 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 3.2 ($\bigsqcup$ 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 3.1, together with an inductive reasoning, ensures 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 3.1, 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.*

It immediately follows from the fact that 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 3.1, $e' \in C$. 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 3.1, $e' \in C_1$ 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' is another lower bound for C_0 and C_1 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 $C' \sqsubseteq C$. ■

In a prime event structure an event e uniquely determines its history, that is the set $\lfloor e \rfloor$ of its causes, independent 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 3.15 (Possible History). Let G be an AES and let $e \in E$. Given a configuration $C \in \mathit{Conf}(G)$ 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 $\mathit{Hist}(e)$, is then defined as

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

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

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

Notice that, by conditions (1) and (2) in the definition of configuration (Definition 3.13), each history $C[\![e]\!]$ is a *finite* set of events. Moreover, each history $C[\![e]\!]$ is characterized by the fact that e is the greatest element with respect to $(\nearrow_{C[\![e]\!]})^*$, and, therefore, for any two events e and e', we have that $\mathit{Hist}(e) \cap \mathit{Hist}(e') \neq \emptyset$ if and only if $e = e'$. It is also easy to see that $(C[\![e]\!])[\![e]\!] = C[\![e]\!]$.

Let us now give some other properties of the set of histories. Point (1) below 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 cannot change after the event has occurred. Point (3) asserts that different histories of the same event are incompatible.

Lemma 3.3 (History Properties). *Let G be an* AES. *Then in $\langle \mathit{Conf}(G), \sqsubseteq\rangle$ we have that*:

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

Proof. 1. Obviously, $C[\![e]\!] \in \mathit{Conf}(G)$. In fact, the requirements (1) and (2) in Definition 3.13 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 3.1, 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

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

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

Lemma 3.4 (Primes). *Let G be an* AES. *Then*

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

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

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

Proof. 1. Let $C \in \mathit{Conf}(G)$ and let $C_0 = \bigsqcup\{C' \in \mathit{Hist}(G) \mid C' \sqsubseteq C\}$, which exists by Lemma 3.2.(1). Then clearly $C_0 \sqsubseteq C$. Moreover for all $e \in C$, by Lemma 3.3.(1), 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 $C[\![e]\!] \in \mathit{Hist}(e)$, for some $e \in E$, be a history and let $A \subseteq \mathit{Conf}(G)$ 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:

$$\begin{aligned} C[\![e]\!] &= (\bigsqcup A)[\![e]\!] && \text{[by Lemma 3.2.(2), since } C[\![e]\!] \sqsubseteq \bigsqcup A\text{]} \\ &= C_e[\![e]\!] && \text{[by Lemma 3.2.(2), since } C_e \sqsubseteq \bigsqcup A\text{]} \\ &\sqsubseteq C_e && \text{[by Lemma 3.2.(1)].} \end{aligned}$$

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

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

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

Since C is 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 3.1 (Configurations Form a Domain). *For any* AES G *the partial order* $\langle Conf(G), \sqsubseteq\rangle$ *is a* (*coherent finitary prime algebraic*) *domain.*

Proof. By Lemma 3.2.(1), $Conf(G)$ is a coherent partial order. By Lemma 3.4, 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 from 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 $e \in C$, a configuration $C_e \in A$ such that $e \in C_e$. Since A is directed and C is finite, the set $\{C_e \mid e \in C\}$ has an upper bound $C' \in A$. Then $C = \bigsqcup_{e \in C} C[\![e]\!] = \bigsqcup_{e \in C} C_e[\![e]\!] \sqsubseteq C'$ follows immediately from Lemma 3.3.(2). 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 Figs. 8a and 8b, at the end of Section 7. In particular notice how asymmetric conflict influences 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. Informally, it states that, as one could expect, we pass from an immediate predecessor of a configuration to the configuration itself by executing a single event.

LEMMA 3.5 (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 $C \sqsubset C \sqcup (C'[\![e']\!]) \sqsubseteq C'$ and thus, by definition of immediate precedence, $C' = C \cup (C'[\![e']\!])$. In the same way $C' = C \cup C'[\![e'']\!]$. Hence, by definition of history, we have $e'(\nearrow_{C'})^* e''(\nearrow_{C'})^* e'$ and thus $e' = e''$ (otherwise $\nearrow_{C'}$ would not be acyclic, contradicting the definition of configuration).

($\Leftarrow$) 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 function between the domains of configurations naturally induced by an AES-morphism is a domain morphism.

LEMMA 3.6 (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 $C_0 \in Conf(G_0)$ 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 in the definition of AES-morphism (Definition 2.5) imply that for all e, e' in C_0 such that $f(e) \neq \bot \neq f(e')$:

$$\lfloor f(e) \rfloor \subseteq f(\lfloor e \rfloor);$$
$$f(e) = f(e') \Rightarrow e = e';$$
$$f(e) \nearrow_1 f(e') \Rightarrow e \nearrow_0 e'.$$

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

Let us now prove that $f^* : Conf(G_0) \to Conf(G_1)$ is a domain morphism. Additivity and stability follow from Lemma 3.2. 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 3.5. ■

Theorem 3.1 and Lemma 3.3 suggest how to define a functor from the category **AES** to the category **Dom**. Instead, the functor going back from **Dom** to **AES** first transforms a domain into a PES via $\mathcal{P}$: **Dom** → **PES**, introduced in Definition 3.2, and then embeds such a PES into **AES** via $\mathcal{J}$: **PES** → **AES**, defined in Proposition 2.1.

DEFINITION 3.16 (From AES's to Domains and Backwards). *The functor $\mathcal{L}_a$:* **AES** → **Dom** *is defined as*:

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

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

- *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 $\mathcal{P}_a$: **Dom** → **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 $\langle Pr(D), \sqsubseteq, \# \rangle$, 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 $\langle Pr(D), \sqsubseteq, \sqsubset \cup \# \rangle$.

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

$$\epsilon_G(C) = e \quad \textit{if } C \in Hist(e),$$

is an AES-*morphism.*

Proof. Observe first that ϵ_G is well-defined since, as noticed before, $Hist(e) \cap Hist(e') = \emptyset$ for $e \neq e'$. Let us verify that ϵ_G satisfies the three conditions imposed on AES-morphisms: for all $C, C' \in Hist(G)$, with $C \in Hist(e)$, $C' \in Hist(e')$:

- $\lfloor \epsilon_G(C) \rfloor \subseteq \epsilon_G(\lfloor C \rfloor)$.

We have:

$$\begin{aligned} &\epsilon_G(\lfloor C \rfloor) = \\ &\quad = \epsilon_G(Pr(C)) \\ &\quad = \epsilon_G(\{C[\![e']\!] \mid e' \in C\}) && \text{[by Lemma 3.4]} \\ &\quad = C \\ &\quad \supseteq \lfloor e \rfloor && \text{[since } C \text{ is left-closed]} \\ &\quad = \lfloor \epsilon_G(C) \rfloor \end{aligned}$$

- $(\epsilon_G(C) = \epsilon_G(C')) \ \wedge \ C \neq C' \Rightarrow C \,\#^a\, C'$.

Let $\epsilon_G(C) = e = e' = \epsilon_G(C')$ and $C \neq C'$. Since $C, C' \in \mathit{Hist}(e)$, by Lemma 3.3.(3), 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 $\mathcal{P}_a(\mathcal{L}_a(G))$.

- $\epsilon_G(C) \nearrow \epsilon_G(C') \Rightarrow C \nearrow C'$.

Let $\epsilon_G(C) = e \nearrow e' = \epsilon_G(C')$. Since the relation $\nearrow$ is irreflexive, surely $e \neq e'$ and thus $C \neq C'$. Now, if $e \not\in C'$ then, by Lemma 3.1, surely $\neg(C \uparrow C')$, thus $C \# C'$ in $\mathcal{P}(\mathcal{L}_a(G))$ and therefore, by definition of $\mathcal{J}$, $C \nearrow C'$ 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 3.2.(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 3.3.(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 $C \nearrow C'$ 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 3.8. *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

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

is also a prime interval in $\mathcal{L}_a(G)$, and, by definition of the functor $\mathcal{E}_a$ (Definition 3.16)

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

Now, by additivity of g and Lemma 2.5.(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 3.5, $g(p) - \bigcup g(Pr(p) - \{p\})$ 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 3.4.(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 3.2 (Coreflection between **AES** and **Dom**). $\mathcal{P}_a \dashv \mathcal{L}_a$.

Proof. Let G be an AES and let $\epsilon_G : \mathcal{P}_a(\mathcal{L}_a(G)) \rightarrow G$ be the morphism defined as in Lemma 3.7. We have to show that given any domain D and AES-morphism $h : \mathcal{P}_a(D) \rightarrow G$, there is a unique domain morphism $g : D \rightarrow \mathcal{L}_a(G)$ 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 $g : D \rightarrow \mathcal{L}_a(G)$ 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 3.6, 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 3.5 we conclude $g(d) \preceq g(d')$.

- *Additive*. Let $X \subseteq D$ be a pairwise compatible set. Then:

$$\begin{aligned}
&g\left(\bigsqcup X\right)\\
&\quad = h^*\left(Pr\left(\bigsqcup X\right)\right)\\
&\quad = h^*\left(\bigcup_{x \in X} Pr(x)\right) \qquad \left[\text{since } Pr\left(\bigsqcup X\right) = \bigcup_{x \in X} Pr(x)\right]\\
&\quad = \bigcup_{x \in X} h^*(Pr(x))\\
&\quad = \bigsqcup_{x \in X} g(x).
\end{aligned}$$

- *Stable*. Let $d, d' \in D$ with $d \uparrow d'$, then:

$$\begin{aligned}
&g(d \sqcap d') =\\
&\quad = h^*(Pr(d \sqcap d'))\\
&\quad = h^*(Pr(d) \cap Pr(d')) \qquad [\text{since } Pr(d \sqcap d') = Pr(d) \cap Pr(d')\\
&\qquad\qquad\qquad\qquad\qquad\quad \text{and } h \text{ injective on } Pr(d) \cup Pr(d')]\\
&\quad = h^*(Pr(d)) \cap h^*(Pr(d'))\\
&\quad = g(d) \sqcap g(d').
\end{aligned}$$

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 3.2 to determine $\mathcal{P}_a(g)(p)$. We have:

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

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 3.8, 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).

$(\mathsf{k} = 1)$ 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, $g'(p) = \{h(p)\} = g(p)$.

$(\mathsf{k} \to \mathsf{k} + 1)$ 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.

Recalling that g and g' are additive, since they coincide on the complete primes of D which is prime algebraic, we conclude that they coincide on the whole domain D. ■

Observe that the above result is, in a sense, modular with respect to some properties of AES's established along this section. Basically it relies on the fact that the configurations of an AES form a domain where the complete prime elements are the possible histories of events and the greatest lower bound and least upper bound of (pairwise) compatible sets are given by set-theoretical intersection and union, respectively. This fact suggests the possibility of extending the results of this section to other classes of event structures, like flow, bundle, or prioritized event structures which should fulfill the mentioned properties.

4. CONTEXTUAL NETS

Contextual nets extend ordinary 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* [26] (or *c-nets* for short) that, following the lines suggested in [20] for C/E systems, add contexts to ordinary P/T nets.

To give the definition of c-net we need some notation for multisets and multirelations. 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 flattening of M, namely the multiset $\sum_{\{a \in A \mid M(a)>0\}} 1 \cdot a$, obtained by changing all nonzero coefficients of M to 1. Sometimes we will confuse the multiset $[\![M]\!] \in \mu A$ with the corresponding subset $\{a \in A \mid M(a) > 0\} \subseteq A$ and use on it the usual set operations and relations. For instance, we say that a multiset M is finite if $[\![M]\!]$, seen as a set, is finite. Conversely, a set $X \subseteq A$ will be sometimes identified with the multiset $\sum_{a \in X} 1 \cdot a$. A *multirelation* $f : A \leftrightarrow B$ is a multiset of $A \times B$. It is called *finitary* if for all $a \in A$ the set $\{b \in B \mid f(a,b) > 0\}$ is finite. The composition of two finitary multirelations $f : A \leftrightarrow B$ and $g : B \leftrightarrow C$ is the (finitary) multirelation $g \circ f : A \leftrightarrow C$ defined as $(g \circ f)(a,c) = \sum_{b \in B} f(a,b) \cdot g(b,c)$. Observe that working with general multirelations the composition may be undefined since infinite coefficients are not allowed. For a multirelation $f : A \leftrightarrow B$ we denote by $\mu f : \mu A \to \mu B$ the (possibly partial) function defined by $\mu f(\sum_{a \in A} n_a \cdot a) = \sum_{b \in B} \sum_{a \in A} (n_a \cdot f(a,b)) \cdot b$ when the summation is well defined and undefined otherwise. Observe that if we think of a multiset $M \in \mu A$ as a multirelation $M : 1 \leftrightarrow A$ (where 1 is any singleton set), then $\mu f(M)$ is the composition of multirelations $f \circ M$, hence the partiality of the function μf. If the multiset M is finite, then $\mu f(M)$ is always defined. When a multirelation $f : A \leftrightarrow B$ satisfies $f(a,b) \leq 1$ for all $a \in A$ and $b \in B$ we sometimes confuse it with the corresponding set-relation and write $f(a,b)$ for $f(a,b) = 1$.

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

Definition 4.17 (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.
- $C \subseteq T \times S$ is a relation, 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 4.18 (Pre-set, Post-set, and Context). Let N be a c-net. 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 finite 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 the set $\underline{A} = \bigcup_{A(t)>0} C(t)$.

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

A different notion of contextual net is conceivable, where the context relation is replaced by a context *multirelation* and the context of transitions is defined as a multiset, rather than a set. We will explain in Section 10 the intuition underlying this different model and how our theory can be extended to cope with it.

A multiset of transitions A is enabled by a marking M if it contains the pre-set of A and, *additionally*, the context of A. Since the context is a set, this formalizes the intuition that a token in a place can be used as context *concurrently* by many transitions.

Definition 4.19 (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 *step relation* between markings is defined as

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

We call $M\,[A\rangle\, M'$ a step. A simple step or a firing is a step involving a single transition, i.e., $M\,[t\rangle\, M'$. A marking M is called *reachable* if there exists a finite step sequence

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

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

starting from the initial marking and leading to M.

Other authors (e.g. [24, 27]) allow for the concurrent firing of transitions that use the same token as context and precondition. For instance, in [24] the formal condition for a multiset A of transitions to be enabled by a marking M is $({}^{\bullet}A \leq M \;\wedge\; \underline{A} \leq M)$. Our definition does not admit such steps, the idea being that concurrent transitions should be allowed to fire also in any order.

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-set, post-set, and context of each transition are “preserved.”

Definition 4.20 (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 $h_S : S_0 \leftrightarrow S_1$ is a *finitary* multirelation such that

1. $\mu h_S(m_0)$ is defined and $\mu h_S(m_0) = m_1$;
2. for each transition $t \in T_0$, $\mu h_S({}^{\bullet}t)$, $\mu h_S(t^{\bullet})$ and $\mu h_S(\underline{t})$ are defined, and

   (i) $\mu h_S({}^{\bullet}t) = {}^{\bullet}\mu h_T(t)$;
   (ii) $\mu h_S(t^{\bullet}) = \mu h_T(t)^{\bullet}$;
   (iii) $\mu h_S(\underline{t}) = \underline{\mu h_T(t)}$.

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

Observe that $\mu h_T(t) = h_T(t)$ when $h_T(t) \neq \bot$, and $\mu h_T(t) = \emptyset$ otherwise. In the last case, by the definition above, the places in the pre-set, post-set, and context of t are forced to be mapped to the empty set; i.e., $\mu h_S({}^{\bullet}t + t^{\bullet} + \underline{t}) = \emptyset$. Furthermore, it is immediate to see that, for any (finite) multiset of transitions $A \in \mu T$, we have that (i) $\mu h_S({}^{\bullet}A) = {}^{\bullet}\mu h_T(A)$, (ii) $\mu h_S(A^{\bullet}) = \mu h_T(A)^{\bullet}$ and (iii) $[\![\mu h_S(\underline{A})]\!] = \underline{\mu h_T(A)}$.

A 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 4.1 (Morphisms Preserve the Token Game). *Let N_0 and N_1 be c-nets, and let $h : N_0 \to N_1$ be a morphism. Then for each $M, M' \in \mu S_0$ and $A \in \mu T_0$*

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

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

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:

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

The seminal work by Winskel [10] presents a coreflection between prime event structures and a subclass of P/T nets, namely *safe* nets. In [14] 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 [9]). In the next sections we will relate by a coreflection (asymmetric and prime) event structures and "semi-weighted c-nets."

DEFINITION 4.21 (Semi-weighted and Safe 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} is a relation (i.e., ${}^\bullet t$ is a set 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 the structure of the net. The relation between safe and semi-weighted contextual nets is further investigated in Section 8, where a more precise comparison of their expressive power is carried out.

5. OCCURRENCE CONTEXTUAL NETS

In the previous section the behaviour of a c-net has been described in a dynamic way, by defining how the token game evolves. Occurrence contextual nets are intended to represent, via the unfolding construction, the behaviour of c-nets in a more static way, by expressing the events (firing of transitions) which can occur in a computation and the dependency relations between them. Occurrence c-nets will be defined as safe c-nets where the dependency relations between transitions satisfy suitable acyclicity and well-foundedness requirements. While for ordinary occurrence nets one has to take into account the causality 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

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

DEFINITION 5.22 (Causality). Let N be a safe c-net. The *causality 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 by $\lfloor x \rfloor$ 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 5.23 (Asymmetric Conflict). Let N be a safe c-net. The *strict asymmetric conflict relation* $\acute{}_N$ is defined as

$$t \acute{}_N t' \quad \text{iff} \quad \underline{t} \cap {}^\bullet t' \neq \emptyset \;\; \text{or} \;\; (t \neq t' \;\wedge\; {}^\bullet t \cap {}^\bullet t' \neq \emptyset).$$

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

$$t \nearrow_N t' \quad \text{iff} \quad t <_N t' \quad \text{or} \quad t \acute{}_N t'.$$

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

$$occur(t, C) \wedge occur(t', C) \;\Rightarrow\; prec_C(t, t'). \qquad (\dagger)$$

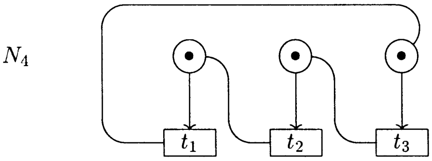


FIG. 5. An occurrence c-net with a cycle of asymmetric conflict.

As suggested by the considerations in the Introduction, in an acyclic safe c-net where any transition is enabled at most once in each computation, condition (†) is surely satisfied when the same place s appears in the context of t and in the pre-set of t'. But (†) 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 symmetric (direct) conflict leads to asymmetric conflicts in both directions. Furthermore, since, as noticed for the abstract model of AES's, (†) is weaker than the condition that expresses causality, the condition (†) is satisfied when t causes (in the usual sense) t'.[7] For technical reasons it is convenient to have a special notation for the strict asymmetric conflict. In the following, 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 Fig. 5 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 $t_1 \nearrow t_2$.

An occurrence c-net is a safe c-net that exhibits an acyclic behaviour and such that each transition can fire in some computation of the net. Furthermore, to allow for the interpretation of the places as token occurrences, each place has at most one transition in its pre-set.

Definition 5.24 (Occurrence c-nets). An *occurrence c-net* is a safe c-net N satisfying the following requirements

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

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

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 the causes of a transition t contain a $\nearrow_N$ cycle then t can never fire, since in an occurrence c-net, 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 mentioned before the asymmetric conflict relation induces a symmetric conflict relation (on sets of transitions) defined in the following way:

Definition 5.25 (Conflict). Let N be a c-net. The *conflict relation* $\# \subseteq \mathbf{2}^T_{\mathit{fin}}$ associated to N is defined as:

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

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

For instance, referring to Fig. 5, we have $\#\{t_1, t_2, t_3\}$, while $\#\{t_i, t_j\}$ does not hold 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 in Definition 5.24.

[7] This is the origin of the weak causality interpretation of $\nearrow$.

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 ordinary 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. Here, due to 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 section introduces such a condition and then shows that it correctly formalizes the intuitive idea of concurrency.

Definition 5.26 (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-set and context is concurrent.

Proposition 5.1. *For any transition t of an occurrence c-net, $conc({}^\bullet t + \underline{t})$.*

Proof. Since $\lfloor {}^\bullet t + \underline{t} \rfloor \cup \{t\} = \lfloor t \rfloor$ conditions (2) and (3) of Definition 5.26 are satisfied by the definition of occurrence c-net. 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, condition (1) is also 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 5.1 (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 $<$ implies $\nearrow$, 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 causally dependent (distinct) places in M'. Since $conc(M - t^\bullet)$, by hypothesis, and $conc({}^\bullet t)$, by Proposition 5.2, 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 5.2 (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 cannot 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 $s \in {}^\bullet t$ (which is included in 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, since $\lfloor M \rfloor$ is finite, also $\lfloor M' \rfloor$ is finite.

Finally, we have to show that there are no (distinct) causally 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 5.2 (Concurrency and Reachability). *Let N be an occurrence c-net and let $M \subseteq S$ be a set of places. Then*

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

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

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

(k = 0) In this case simply $m \supseteq M$ and thus $m = M^{(0)} \supseteq M$.

(k > 0) By construction, $t^{(k)}$ is maximal in $\lfloor M \rfloor$ with respect to $(\nearrow_{\lfloor M \rfloor})^+$. Thus, by Lemma 5.1, 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

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

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

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

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

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

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 5.2 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 5.1. *For any transition t of an occurrence c-net N there is a reachable marking M of N which enables t.*

Proof. By Proposition 5.1, $conc({}^\bullet t + \underline{t})$ and thus, by Proposition 5.2, we can find a reachable marking M of N, such that $M \supseteq {}^\bullet t + \underline{t}$, enabling t. ■

5.3. Morphisms on Occurrence Contextual Nets

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

LEMMA 5.3 (Occurrence c-nets Morphisms). *Let N_0 and N_1 be occurrence c-nets and let $h : N_0 \to N_1$ 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 $t_0 \in T_0$ and $t_1 \in T_1$, 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 $s_0 \in S_0$, $s_1 \in S_1$, $t_1 \in T_1$:

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

Proof (Sketch). 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 $s_0 \in m_0$ 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 section, c-net morphisms preserve the concurrency relation.

COROLLARY 5.2 (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 5.2, 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 4.1, 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 5.2, *conc*$(\mu h_S(M_0))$. ■

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

In the next theorem we show that, more generally, morphisms preserve the "amount of concurrency," namely they reflect causality and conflict, while asymmetric conflict is reflected or becomes conflict. The fact that asymmetric conflict is not necessarily reflected is related to the fact that the asymmetric conflict relation for an occurrence c-net does not satisfy the saturation condition required for AES's (see Definition 2.4).

THEOREM 5.1. *Let N_0 and N_1 be occurrence c-nets and let $h : N_0 \to N_1$ be a morphism. Then, for all $t_0, t_0' \in T_0$ such that $h_T(t_0) \neq \bot \neq h_T(t_0')$*

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

Proof. 1. Let the symbol $\prec$ denote the immediate causal dependency between transitions, namely $t \prec t'$ if $t < t'$ and there does not exist t'' such that $t < t'' < t'$. The desired property easily follows by observing that c-net morphisms reflect $\prec$-chains, namely that if $t_1^{(0)} \prec t_1^{(1)} \prec \cdots \prec t_1^{(n)}$ is a chain of transitions in N_1 such that $t_1^{(n)} = h_T(t_0^{(n)})$, then there exists a chain $t_0^{(0)} \prec t_0^{(1)} \prec \cdots \prec t_0^{(n)}$ in N_0

such that $t_1^{(i)} = h_T(t_0^{(i)})$ for all $i \in \{0, \dots, n\}$. This fact can be proved by induction on n, exploiting Lemma 5.3.

2. Let $h_T(t_0) = h_T(t_0')$ and $t_0 \neq t_0'$. Consider a chain of transitions $t_1^{(0)} \prec \cdots \prec t_1^{(k)} = h_T(t_0)$ such that ${}^\bullet t_1^{(0)} \subseteq m_1$ and $t_1^{(i)\bullet} \cap {}^\bullet t_1^{(i+1)} \neq \emptyset$ for all $i \in \{0, \dots, k-1\}$ (the existence of such a finite chain is an immediate consequence of the definition of occurrence c-net). Since, as observed in point (1), morphisms reflects $\prec$-chains, there are in T_0 two $\prec$-chains of transitions,

$$t_0^{(0)} \prec \cdots \prec t_0^{(k)} \qquad \text{and} \qquad t_0^{\prime(0)} \prec \cdots \prec t_0^{\prime(k)},$$

such that, $h_T(t_0^{(i)}) = h_T(t_0^{\prime(i)}) = t_1^{(i)}$, for all $i \in \{1, \dots, k\}$ and $t_0 = t_0^{(k)}$, $t_0' = t_0^{\prime(k)}$.

Let j be the least index such that $t_0^{(j)} \neq t_0^{\prime(j)}$. 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 $s_0' \in {}^\bullet t_0^{\prime(0)}$ such that $h_S(s_0, s_1)$ and $h_S(s_0', s_1)$. By Lemma 5.3, since $s_1 \in m_1$, also s_0 and s_0' are in the initial marking and thus $s_0 = s_0'$. Hence $t_0^{(0)} \nearrow_0 t_0^{\prime(0)} \nearrow_0 t_0^{(0)}$ and thus, 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.

3. We distinguish two cases. If $h_T(t_0) \leadsto_1 h_T(t_0')$ 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 $t_0 \leadsto_1 t_0'$. Otherwise s_0 and s_0' are in the post-sets of two transitions $t_0^{(0)}$ and $t_0^{\prime(0)}$, which are mapped to the same transition in N_1 (the transition which has s_1 in its post-set). By point (2), $t_0^{(0)}$ and $t_0^{\prime(0)}$ are identical or in conflict: in the first case $s_0 = s_0'$ and thus $t_0 \leadsto_0 t_0'$, while in the second case $t_0 \#_0 t_0'$.

If, instead, $h_T(t_0) <_1 h_T(t_0')$, then, by point (1), 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 (2) that either $t_0'' = t_0$ and thus $t_0 <_0 t_0'$, or $t_0'' \#_0 t_0$ and thus $t_0 \#_0 t_0'$.

4. Recall that if $\# h_T(A)$ then $\lfloor h_T(A) \rfloor$ contains a cycle of asymmetric conflict. Now, by point (1), $\lfloor h_T(A) \rfloor \subseteq h_T(\lfloor A \rfloor)$ and thus, by point (3), it is easy to conclude the thesis. ∎

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 ordinary nets, each transition in $\mathcal{U}_a(N)$ represents a 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 $f_N : \mathcal{U}_a(N) \rightarrow N$, called the folding morphism. The unfolding operation can be extended to a functor $\mathcal{U}_a$: **SW-CN** → **O-CN** that is right adjoint to the inclusion functor $\mathcal{I}_{oc}$: **O-CN** → **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 $t \in T_0$, and $i_S(s, s') = 1$ if $s = s'$ and 0 otherwise, for $s, s' \in S_0$) 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 6.27 (Depth). Let N be an occurrence c-net. The function *depth* : $S \cup T \rightarrow \mathbb{N}$ is defined inductively as follows:

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

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 than 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 6.1. *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 which is enabled by (the image of) a concurrent subset of places in the partial unfolding currently generated. For technical reasons we prefer to give an equivalent axiomatic definition.

Definition 6.28 (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:

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

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

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

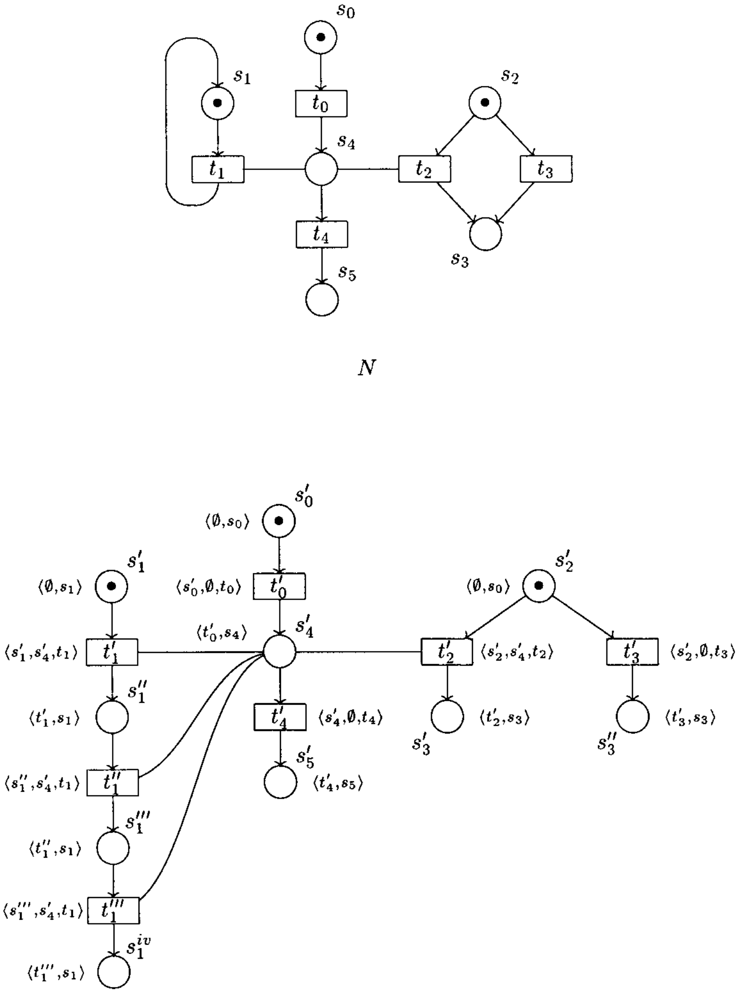
Places and transitions in the unfolding of a c-net represent, respectively, tokens and firing of transitions in the original net. Each place in the unfolding is a pair recording the "history" of the token and the corresponding place in the original net. Each transition is a triple recording the pre-set 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 m of N (recall that m is a set since N is semi-weighted). 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 $M_p + M_c$ that corresponds, in the original net, to a marking that enables t (M_p corresponds to the pre-set and M_c to the context used by t). For each place s in the post-set of such a transition 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 6 shows a c-net N and an 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, enriched with a superscript. The figure also reports the concrete identity of the items of the unfolding.

Occurrence c-nets are particular semi-weighted c-nets and thus we can consider the inclusion functor $\mathcal{I}_{oc}$: **O-CN** → **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 $\mathcal{I}_{oc}$ and thus establishes a coreflection between **SW-CN** and **O-CN**.

Theorem 6.1 (Coreflection between **SW-CN** and **O-CN**). $\mathcal{I}_{oc} \dashv \mathcal{U}_a$.

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 6.28. 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:

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


s_0
s_1
t_0
s_2
s_4
t_1
t_2
t_3
t_4
s_3
s_5
s'_0
$\langle \emptyset, s_0 \rangle$
s'_1
$\langle \emptyset, s_1 \rangle$
$\langle s'_0, \emptyset, t_0 \rangle$
t'_0
s'_2
$\langle \emptyset, s_0 \rangle$
$\langle t'_0, s_4 \rangle$
s'_4
$\langle s'_1, s'_4, t_1 \rangle$
t'_1
t'_2
$\langle s'_2, s'_4, t_2 \rangle$
t'_3
$\langle s'_2, \emptyset, t_3 \rangle$
s''_1
$\langle t'_1, s_1 \rangle$
t'_4
$\langle s'_4, \emptyset, t_4 \rangle$
$\langle t'_2, s_3 \rangle$
$\langle t'_3, s_3 \rangle$
s'_3
s''_3
s'_5
$\langle s''_1, s'_4, t_1 \rangle$
t''_1
$\langle t'_4, s_5 \rangle$
s'''_1
$\langle t''_1, s_1 \rangle$
$\langle s'''_1, s'_4, t_1 \rangle$
t'''_1
s^{iv}_1
$\langle t'''_1, s_1 \rangle$

N

$\mathcal{U}_a(N)^{[3]}$

FIG. 6. A c-net and (a part of) its unfolding.

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

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

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

($\mathsf{k} = 0$) 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:

$$h_T{}^{[0]} = \emptyset,$$

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

($\mathsf{k} \to \mathsf{k} + 1$) 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 $depth(t_1) = k + 1$. By definition of depth, $depth(s) \leq k$ 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 $h_T{}^{[k+1]}(t_1) = \bot$ and $h_S{}^{[k+1]}(s_1, s') = 0$ for all $s_1 \in t_1{}^\bullet$ and $s' \in S'$.

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

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

Since N_1 is an occurrence c-net, ${}^\bullet t_1 \cap \underline{t_1} = \emptyset$ and, by Proposition 5.2, $conc({}^\bullet t_1 \cup \underline{t_1})$. Hence, by Corollary 5.2,

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

Moreover, by construction, $f_N \circ h^{[k]} = g_{|{N_1}^{[k]}}$, and therefore

$$\mu f_S(M_p) = \mu f_S\big(\mu {h_S}^{[k]}({}^\bullet t_1)\big) = \mu g_S({}^\bullet t_1) = {}^\bullet t,$$

where the last passage is justified by the definition of c-net morphism, and in the same way $\mu f_S(M_c) = \underline{t}$. Thus, by the 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 must 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 check allows us 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 $h' : N_1 \to \mathcal{U}_a(N)$ be 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:

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

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

($\mathsf{k \to k + 1}$) 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 $M_c = \underline{t'} = \mu h'_S(\underline{t_1})$. 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 CONTEXTUAL 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 can obtain a pre-AES by simply forgetting the places and remembering the dependency relations that they induce between transitions, namely causality and asymmetric conflict. The corresponding (saturated) AES has the same causal relation $\leq_N$, while asymmetric conflict is given by the union of asymmetric conflict $\nearrow_N$ and of the induced binary conflict $\#_N$. Furthermore a morphism between occurrence c-nets naturally restricts to a morphism between the corresponding AES's.

Definition 7.29 (From Occurrence c-nets to AES's). Let $\mathcal{E}_a : \textbf{O-CN} \rightarrow \textbf{AES}$ be the functor defined as:

- for each occurrence c-net N, if $\#_N$ denotes the induced binary conflict in N:
$$\mathcal{E}_a(N) = \langle T, \leq_N, \nearrow_N \cup \#_N \rangle;$$
- for each morphism $h : N_0 \rightarrow N_1$:
$$\mathcal{E}_a(h : N_0 \rightarrow N_1) = h_T.$$

Notice that the induced conflict relation $\#^a$ in the AES $\mathcal{E}_a(N)$ (see Definition 2.3) coincides with the induced conflict relation in the net N (see Definition 5.25). Therefore in the following we will confuse the two relations and simply write # to denote both of them.

Proposition 7.1 (Well-definedness). *$\mathcal{E}_a$ is a well-defined functor.*

Proof. Given any occurrence c-net N, by Definition 5.24 and the considerations on the saturation of pre-AES's following Definition 2.4, we immediately have that $\mathcal{E}_a(N)$ is an AES. Furthermore, if $h : N_0 \rightarrow N_1$ is a c-net morphism, then, by Theorem 5.1, $\mathcal{E}_a(h) = h_T$ is an AES-morphism. Finally $\mathcal{E}_a$ obviously preserves arrow composition and identities. ■

To go the other way around, from an AES we can obtain a canonical occurrence c-net via a free construction that mimics Winskel's. In the constructed c-net the events are used as transitions, and for each set of events related in a certain way by causality and asymmetric conflict, a unique place is generated that induces such kind of relations on the corresponding transitions.

Definition 7.30 (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:

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

As anticipated, 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 and a context for all the events in A. Moreover, if $x = \{e\}$, such a 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 7 presents some examples of basic AES's with the corresponding c-nets. The cases of an AES with two events related, respectively, by causality, asymmetric conflict, and (immediate symmetric)

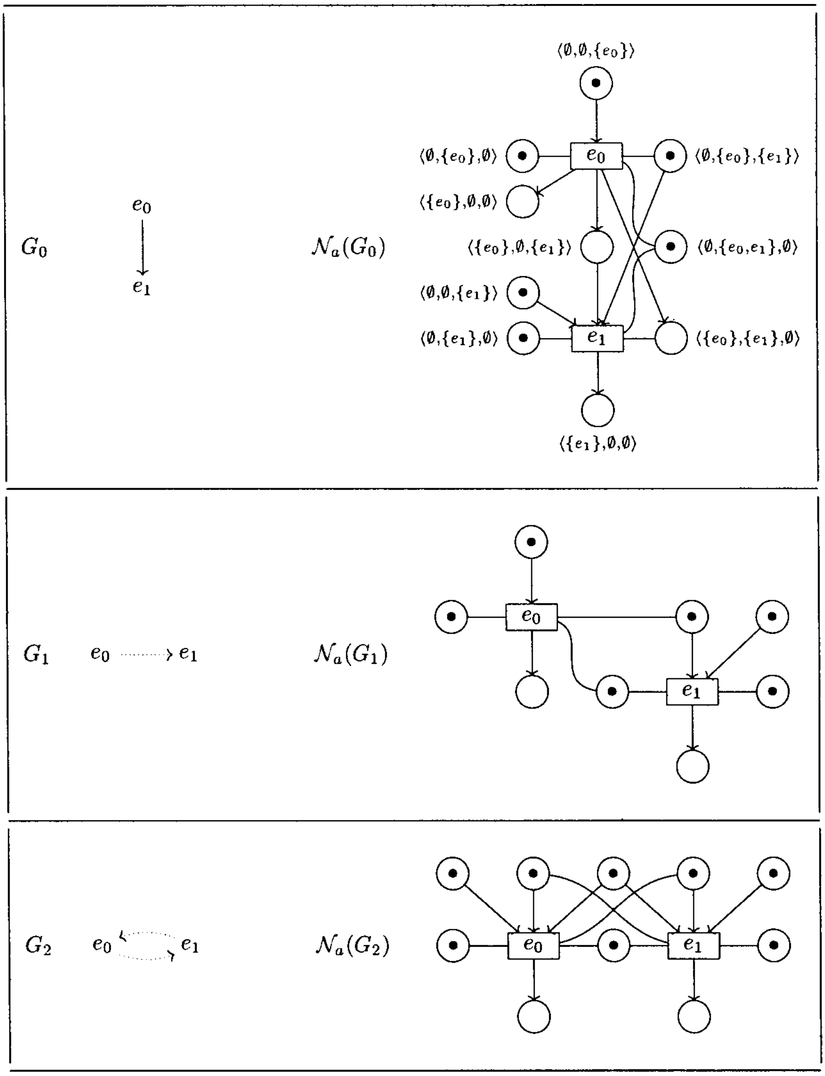


FIG. 7. Three simple AES's and the corresponding occurrence c-nets produced by the functor $\mathcal{N}_a$.

conflict are considered. Pictorially, an asymmetric conflict $e_0 \nearrow e_1$ is represented by a dotted arrow from e_0 to e_1. Causality is represented, as usual, by plain arrows. In the first case the places of the net are annotated with their concrete identity.

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

LEMMA 7.1. *Let $G = \langle E, \leq, \nearrow \rangle$ be an* AES *and let $\mathcal{N}_a(G)$ be the c-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* $e \nearrow e'$;
3. $e \nearrow_N e'$ *iff* $e \nearrow e'$.

Proof. 1. Let $\prec_N$ denote the immediate causality relation in N. 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 $\mathcal{N}_a$, $e < e'$. In contrast, if $e < e'$ then $\langle \{e\}, \emptyset, \{e'\} \rangle \in S$ and thus $e \prec_N e'$. Since $<_N$ is the transitive closure of $\prec_N$ and $<$ is a transitive relation we conclude the thesis.

2. If $e{}^{-}_{N}e'$ then there exists a place $\langle x, A, B\rangle \in S$ with $e \in A \cup B$ and $e' \in B$ and thus either $e \nearrow e'$ or $e \# e'$. But since G is an AES, the binary conflict is included in the asymmetric conflict and thus, also in the second case, $e \nearrow e'$. In contrast, if $e \nearrow e'$ then $\langle \emptyset, \{e\}, \{e'\}\rangle \in S$ and thus $e \rightsquigarrow_N e'$.

3. Easy consequence of points (1) and (2). ■

As an immediate corollary we have:

Corollary 7.1. *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. By Lemma 7.1 the causality relation $\leq_N \; = \; \leq$ and the asymmetric conflict $\nearrow_N \; = \; \nearrow$ inherits the necessary properties from those of G. ■

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 later simplify the notation).

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

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 7.2. *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 ${}^\bullet s_0 = \{t_0\}$.[8] Therefore $h_T(t_0) = t_1$ is defined (see the remark after Definition 4.20) 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 Theorem 5.3.(2), we conclude that s_0' and s_0 necessarily coincide, otherwise they would be in the post-set 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). ■

Recall that, by Lemma 7.1, for any AES $G = \langle E, \leq, \nearrow\rangle$ the causality and asymmetric conflict relations in $\mathcal{N}_a(G)$ coincide with $\leq$ and $\nearrow$. Hence $\mathcal{E}_a(\mathcal{N}_a(G)) = \langle E, \leq, \nearrow'\rangle$, with $\nearrow' = \nearrow \cup \# = \nearrow$, where the last equality is justified by the fact that in an AES $\# \subseteq \nearrow$. Hence $\mathcal{E}_a \circ \mathcal{N}_a$ is the identity on objects.

We next prove that $\mathcal{N}_a$ extends to a functor from **AES** to **O-CN**, which is left adjoint to $\mathcal{E}_a$ (with unit the identity id_G). More precisely they establish a coreflection between **AES** and **O-CN**.

Theorem 7.1 (Coreflection between **O-CN** and **AES**). $\mathcal{N}_a \dashv \mathcal{E}_a$.

Proof. Let $G = \langle E, \leq, \nearrow\rangle$ be an AES and let $\mathcal{N}_a(G) = \langle S, T, F, C, m\rangle$ be as in Definition 7.30. 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

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

unique morphism $h : \mathcal{N}_a(G) \to N_0$, such that the following diagram commutes:

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

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

$$h_T = g.$$

Therefore we only have to show that a multirelation $h_S : S \leftrightarrow S_0$ such that $\langle h_T, h_S\rangle$ is a c-net morphism exists and it is uniquely determined by h_T.

Existence. Let us define h_S in such a way it satisfies the conditions of Lemma 7.2, 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, $s_0 \in m_0$. In contrast, let $s_0 \in m_0$ and let

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

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 $t \nearrow t'$ 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 $x = h_T^{-1}({}^\bullet s_0) \cap \lfloor t \rfloor$. Since N_0 is an occurrence c-net $\mid {}^\bullet s_0 \mid \leq 1$ and thus $\mid x \mid \leq 1$ (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 c-net morphism.

Uniqueness. The multirelation h_S such that $\langle h_T, h_S\rangle$ is a c-net morphism is unique essentially because it is completely determined by the conditions of Lemma 7.2. More precisely, if $h'_S : S \leftrightarrow S_0$ is another multirelation such that $\langle h_T, h'_S\rangle$ is a morphism and $h'_S(s, s_0)$, then necessarily by Lemma 7.2, $h_S(s, s_0)$. 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 7.2 and the 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 allows us to conclude when $x = \{t\}$. ■

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 $\langle \{e_1\}, A_1, B_1\rangle$

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

As mentioned before, once we have an AES semantics for contextual nets, the coreflection between **AES** an **Dom** (Theorem 3.2) immediately provides a domain semantics. Then, the equivalence between **PES** and **Dom** (see Section 3.1) can be used to "translate" the domain semantics of semi-weighted c-nets into a prime event structure semantics. This completes the following chain of coreflections between **SW-CN** and **PES**:

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

Figure 8 shows (a part of) the AES, the domain, and the PES associated to the c-net of Fig. 6. 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, as discussed in the Introduction. For instance, the PES semantics of the net N_3 in Fig. 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 Fig. 9 have the same PES semantics, but different AES semantics.

8. RELATION WITH WINSKEL'S SEMANTICS FOR ORDINARY NETS

In this section we study the relationship between the proposed semantics for semi-weighted contextual nets and the classical Winskel's semantics for safe ordinary nets (generalized to semi-weighted ordinary nets in [31]). Then, we formally compare the expressiveness of semi-weighted and safe contextual nets by resorting to their prime event structure semantics.

Let us start by considering the diagram in Fig. 10. The top row represents the chain of coreflections defined in [14, 31], leading from the category **SW-N** of semi-weighted ordinary nets to the category **Dom**, through the category **O-N** of occurrence nets. In the mentioned paper it is shown that such coreflections restrict, for safe nets, to Winskel's coreflections. The bottom row, instead, summarizes our coreflective semantics for contextual nets. The vertical functors $\mathcal{I}_{nc}$: **SW-N** → **SW-CN** and $\mathcal{I}_{nco}$: **O-N** → **O-CN** are inclusions, while $\mathcal{J}$: **PES** → **AES** is the full embedding functor introduced in Proposition 2.1. We want to show that, as suggested by some previous informal considerations, each of our coreflections cuts down to Winskel's coreflection between the corresponding subcategories.

Let us first concentrate on square (1). It is easy to see that the unfolding functor $\mathcal{U}_a$ restricts to $\mathcal{U}$ in the sense that $\mathcal{I}_{nco} \circ \mathcal{U} = \mathcal{U}_a \circ \mathcal{I}_{nc}$. Similarly, the inclusion $\mathcal{I}_{oc}$ restricts to $\mathcal{I}_o$; i.e., $\mathcal{I}_{nc} \circ \mathcal{I}_o = \mathcal{I}_{oc} \circ \mathcal{I}_{nco}$.

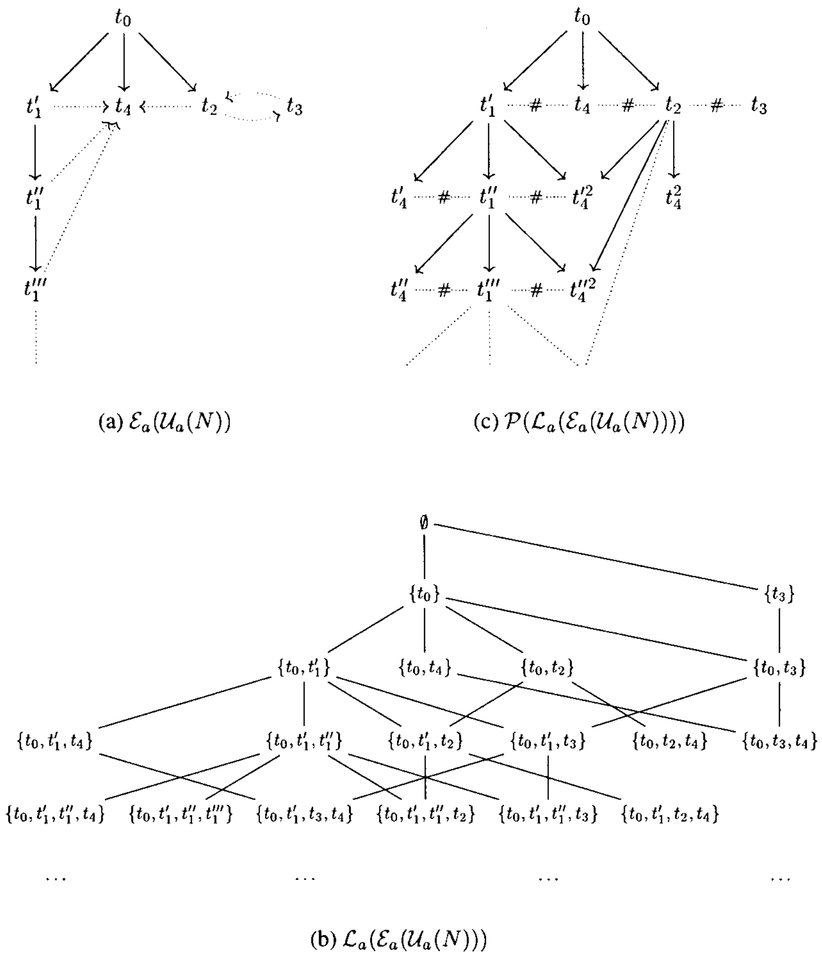


FIG. 8. The (a) AES, (b) domain, and (c) PES for the c-net N of Fig. 6.

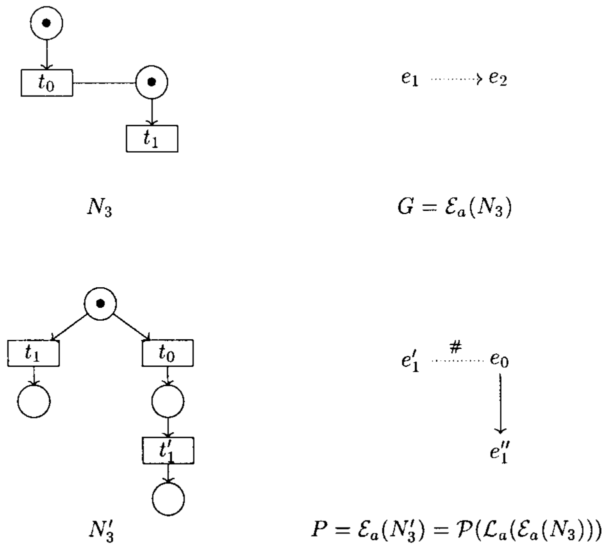


FIG. 9. AES semantics is finer than PES semantics.

$$\begin{tikzcd}[column sep=large, row sep=large]
\textbf{SW-N} \arrow[r, shift right, "\mathcal{U}"'] \arrow[r, phantom, "\perp"] \arrow[dd, hook, "\mathcal{I}_{nc}"'] & \textbf{O-N} \arrow[l, hook', shift right, "\mathcal{I}_o"'] \arrow[r, shift right, "\mathcal{E}"'] \arrow[r, phantom, "\perp"] \arrow[dd, hook, "\mathcal{I}_{nco}"'] & \textbf{PES} \arrow[l, shift right, "\mathcal{N}"'] \arrow[dd, hook, "\mathcal{J}"'] \arrow[dr, shift right, "\mathcal{L}"'] \arrow[dr, phantom, "\perp"] & \\
(1) & (2) & (3) & \textbf{Dom} \arrow[ul, shift right, "\mathcal{P}"'] \arrow[dl, shift right, "\mathcal{P}_a"'] \\
\textbf{SW-CN} \arrow[r, shift right, "\mathcal{U}_a"'] \arrow[r, phantom, "\perp"] & \textbf{O-CN} \arrow[l, hook', shift right, "\mathcal{I}_{oc}"'] \arrow[r, shift right, "\mathcal{E}_a"'] \arrow[r, phantom, "\perp"] & \textbf{AES} \arrow[l, shift right, "\mathcal{N}_a"'] \arrow[ur, shift right, "\mathcal{L}_a"'] \arrow[ur, phantom, "\perp"] &
\end{tikzcd}$$

.

FIG. 10. Relating the semantics of ordinary and contextual nets.

Since the inclusions $\mathcal{I}_{nc}$ and $\mathcal{I}_{nco}$ are full embeddings, by general categorical arguments, from the fact that $\mathcal{U}_a \vdash \mathcal{I}_{oc}$ is a coreflection we immediately conclude that $\mathcal{U} \vdash \mathcal{I}_o$ and that such adjunction is a coreflection as well. A similar reasoning applies to the "degenerate" square (3) (we can imagine its right edge to be the identity functor on **Dom**), just observing that $\mathcal{J} \circ \mathcal{P} = \mathcal{P}_a$ and $\mathcal{L}_a \circ \mathcal{J} = \mathcal{L}$.

When considering square (2) instead, the correspondence is not completely straightforward. The vertical "edges" of the square, namely $\mathcal{I}_{nco}$ and $\mathcal{J}$, are still full embedding functors and $\mathcal{J} \circ \mathcal{E} = \mathcal{E}_a \circ \mathcal{I}_{nco}$, but the other commutativity property, i.e., $\mathcal{I}_{nco} \circ \mathcal{N} = \mathcal{N}_a \circ \mathcal{J}$ fails to hold. In fact, given a PES P, the net $\mathcal{I}_{nco}(\mathcal{N}(P))$ is obtained by saturating P with places acting as preconditions and postconditions for the events in P, while in $\mathcal{N}_a(\mathcal{J}(P))$ also context places are added. In this case we resort to the following categorical result which generalizes the observation used for the other two squares.

LEMMA 8.1. *Let* $\mathbf{A}_i$ *and* $\mathbf{B}_i$ *for* $i \in \{0, 1\}$ *be categories, let* $F_i : \mathbf{A}_i \to \mathbf{B}_i$, $G_i : \mathbf{B}_i \to \mathbf{A}_i$ *be functors, and let* $I_A : \mathbf{A}_0 \to \mathbf{A}_1$, $I_B : \mathbf{B}_0 \to \mathbf{B}_1$ *be full embedding functors* (*see Fig.* 11). *Suppose that*

1. $F_1 \vdash G_1$;
2. $F_1 \circ I_A = I_B \circ F_0$;
3. *there is a natural transformation* $\alpha : G_1 \circ I_B \to I_A \circ G_0$, *such that for all objects* A *in* $\mathbf{A}_0$ *and* B *in* $\mathbf{B}_0$, *each arrow* $g : G_1(I_B(B)) \to I_A(A)$ *uniquely factorizes through* α_B, *i.e., there exists a unique* $f : I_A(G_0(B)) \to I_A(A)$ *such that* $g = f \circ \alpha_B$

$$\begin{tikzcd}
G_1(I_B(B)) \arrow[r, "\alpha_B"] \arrow[dr, "g"'] & I_A(G_0(B)) \arrow[d, dotted, "f"] \\
& I_A(A)
\end{tikzcd}$$

Then $F_0 \vdash G_0$. *Furthermore if the units of* $F_1 \vdash G_1$ *and* $F_1 \circ \alpha$ *are natural isomorphisms then so is the unit of* $F_0 \vdash G_0$ *as well.*

Proof (Sketch). Let $\eta^1 : 1 \to F_1 \circ G_1$ be the unit of the adjunction $F_1 \vdash G_1$. Given an object B in $\mathbf{B}_0$, consider the arrow $F_1(\alpha_B) \circ \eta^1_{I_B(B)} : I_B(B) \to I_B(F_0(G_0(B)))$

$$I_B(B) \xrightarrow{\eta^1_{I_B}(B)} F_1(G_1(I_B(B))) \xrightarrow{F_1(\alpha B)} F_1(I_A(G_0(B))) = I_B(F_0(G_0(B))).$$

Then one can prove that $F_0 \vdash G_0$ with unit $\eta^0_B = I_B^{-1}(F_1(\alpha_B) \circ \eta^1_{I_B(B)})$. ∎

Coming back to square (2), observe that there is a natural transformation $\alpha : \mathcal{N}_a \circ \mathcal{J} \to \mathcal{I}_{nco} \circ \mathcal{N}$, which essentially forgets the contexts. The component at a PES $P = \langle E, \leq, \# \rangle$ of α is given by $\alpha_P = \langle id_E, \alpha_{P_S} \rangle : \mathcal{N}_a(\mathcal{J}(P)) \to \mathcal{I}_{nco}(\mathcal{N}(P))$, where α_{P_S} is a partial function defined, for any place s in the contextual net $\mathcal{N}_a(\mathcal{J}(P))$, as follows:

$$\alpha_{P_S}(s) = \begin{cases} \bot & \text{if } s \text{ is a context place for some transition } t \\ s & \text{otherwise.} \end{cases}$$

Furthermore, given any PES P and (ordinary) occurrence net N, each arrow $g : \mathcal{N}_a(\mathcal{J}(P)) \to \mathcal{I}_{nco}(N)$

$$\begin{tikzcd}
\mathbf{A}_0 \arrow[r, shift right=1.5, "F_0"'] \arrow[r, phantom, "\perp"] \arrow[d, hook, "I_A"'] & \mathbf{B}_0 \arrow[l, shift right=1.5, "G_0"'] \arrow[d, hook, "I_B"] \\
\mathbf{A}_1 \arrow[r, shift right=1.5, "F_1"'] \arrow[r, phantom, "\perp"] & \mathbf{B}_1 \arrow[l, shift right=1.5, "G_1"'] \arrow[ul, Rightarrow, "\alpha"']
\end{tikzcd}$$

FIG. 11. Restriction of an adjunction.

can be factorized uniquely as $f \circ \alpha_P$, for $f : \mathcal{I}_{nco}(\mathcal{N}(P)) \to \mathcal{I}_{nco}(N)$:

$$\begin{tikzcd}
\mathcal{N}_a(\mathcal{J}(P)) \arrow[r, "\alpha_P"] \arrow[dr, "g"'] & \mathcal{I}_{nco}(\mathcal{N}(P)) \arrow[d, dotted, "f"] \\
& \mathcal{I}_{nco}(N).
\end{tikzcd}$$

In fact, since the transitions in $\mathcal{I}_{nco}(N)$ have an empty context, necessarily g must map the context places in $\mathcal{N}_a(\mathcal{J}(P))$ to the empty multiset, and thus f is uniquely determined as the restriction of g to $\mathcal{I}_{nco}(\mathcal{N}(P))$. Finally, it is easy to verify that $\mathcal{E}_a \circ \alpha$ is a natural isomorphism. Hence we can apply Lemma 8.1 to conclude that our coreflection $\mathcal{E}_a \vdash \mathcal{N}_a$ induces the coreflection $\mathcal{E} \vdash \mathcal{N}$.

Let us now comment on the expressiveness of semi-weighted and safe contextual nets by exploiting the proposed event structure semantics as a formal means to compare the two classes of nets. As discussed in the Introduction, in the case of ordinary nets the safeness condition prevents one to model an unbounded degree of concurrency. Formally, in the PES semantics of a finite safe net N the cardinality of a concurrent set of events is bounded by the number of transitions in N; the same applies to finite safe *contextual* nets as well. Instead, observe that the PES semantics of the semi-weighted c-net N_4 of Fig. 12 includes sets of concurrent events of unbounded cardinality, namely all finite subsets of $f_{N_4}^{-1}(t_1)$, where $f_{N_4} : \mathcal{U}_a(N_4) \to N_4$ is the folding morphism. Even more interestingly, let us first recall that, as proved in [20], any finite safe contextual net can be translated into a finite safe ordinary net, having the same process semantics and thus, a fortiori, the same PES semantics. Instead there is no finite general (ordinary) P/T net having the same PES semantics as N_4. In fact, in the PES associated to any P/T net, the number of events which are directly caused by a single event e is bounded by the number of tokens produced by the transition corresponding to e. Instead, in the PES associated to N_4 the event corresponding to t_2 is an immediate cause of infinitely many other events (all the events corresponding to transition t_1).

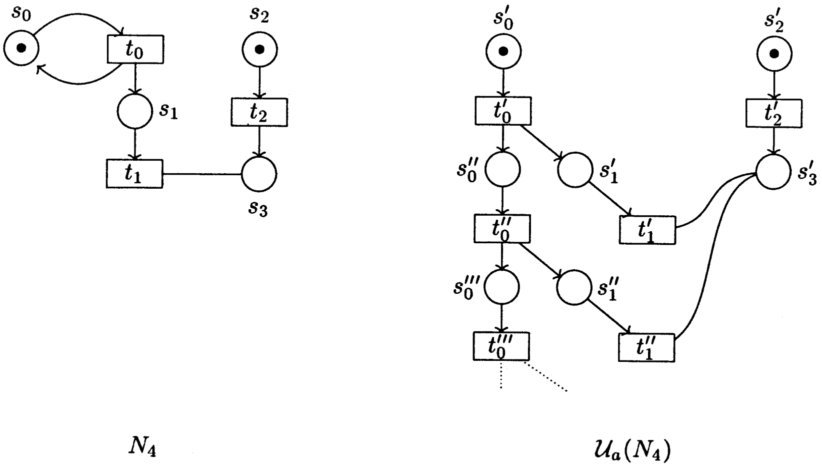


FIG. 12. A semi-weighted contextual net N_4 and (a part of) its unfolding where a transition occurrence directly causes infinitely many other transition occurrences.

9. 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 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, such as [8, 26, 29, 39]. In her Ph.D. thesis [3], Busi 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 [11, 37].[9]

The papers [26, 29, 39] extend the theory of concatenable processes of ordinary nets [4] 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 [29] a purely algebraic characterization of such a category is given.

Since the category $\mathbf{CP}[N]$ of concatenable processes of a net N provides a computational model for N, expressing its operational behaviour, we are naturally lead to compare such semantics with the one based on the unfolding, proposed in our paper. In this section, relying on the notion of concatenable c-net process and exploiting the chain of coreflections from **SW-CN** to **Dom**, we establish a close relationship between process and unfolding semantics for c-nets. More precisely, we generalize to c-nets (in the semi-weighted case) a result proved in [9] for ordinary nets, stating that the domain associated to a semi-weighted net N (in our case $\mathcal{L}_a(\mathcal{E}_a(\mathcal{U}_a(N)))$) coincides with the completion of the preorder obtained as the comma category of $\mathbf{CP}[N]$ under 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.

9.1. Contextual Net 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 with at most 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 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 its 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 9.31 (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 4.20, but (1)).

Equivalently, if we denote by $\mathbf{CN}^*$ the subcategory of **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

[9] A different notion of enabling allowing for the simultaneous firing of weakly dependent transitions is used in [11], making difficult a complete direct comparison. For the same reason, although "syntactically" the processes of [8] coincide with ours, they are intended to represent the same firing sequences, but different step sequences.

CN*.[10] This gives also the (obvious) notion of isomorphism between processes, which is an isomorphism 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. 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 [5]. 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 9.32 (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 a finite 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 9.33 (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 by ${}^\bullet\pi$ and $\pi^\bullet$ the multisets $\mu\pi_S(\min(\pi))$ and $\mu\pi_S(\max(\pi))$, 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.

9.2. Concatenable Processes

As in [29, 39] a notion of concatenable process for contextual nets, endowed with an operation of sequential (and parallel) composition, can be easily defined, generalizing the concatenable processes of [4]. Obviously, a meaningful operation of sequential composition can be defined only on the unmarked processes of a c-net. In order to properly define such an operation we need to impose a suitable ordering over the places in $\min(\pi)$ and $\max(\pi)$ for each process π. Such ordering allows us to distinguish among "interface" places of O_π which are mapped to the same place of the original net, a capability which is essential to make sequential composition consistent with the causal dependencies.

Definition 9.34. 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 $\mathbb{N}$ that it naturally induces (formally defined as $\bigcup_{b\in B} \alpha_b$).

Definition 9.35 (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 $\max(\pi)$.

[10] Recall that given a category **C** and an object x of **C**, the *comma category of objects (of* **C***) over* x, denoted $(\mathbf{C} \downarrow x)$, has arrows $f : y \to x$ in **C** as objects. Moreover, given $f : y \to x$ and $g : z \to x$, an arrow $k : f \to g$ in $(\mathbf{C} \downarrow x)$ is an arrow $k : y \to z$ in **C** such that $f = g \circ k$. Symmetrically, the *comma category of objects (of* **C***) under* x, denoted $(x \downarrow \mathbf{C})$, has arrows $f : x \to y$ in **C** as objects. Furthermore, given $f : x \to y$ and $g : x \to z$, an arrow $k : f \to g$ in $(x \downarrow \mathbf{C})$ is an arrow $k : y \to z$ in **C** such that $k \circ f = g$.

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 9.36 (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 the 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 seen at abstract level, namely $[\delta_1; \delta_2']$, does not depend on the particular choice of the representatives.

Definition 9.37 (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. Since such an operation is not relevant for our present aim, we refer the interested reader to [29, 39].

9.3. 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 a 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 9.1. *For any c-net $N = \langle S, T, F, C, m \rangle$ 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 following 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 we write $\delta_1 \lesssim \delta_2$ 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 following. It essentially formalizes the intuitive idea that 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 9.38 (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 deterministic 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 9.2. *Let $\delta_i : m \to M_i$ $(i \in \{1, 2\})$ be two objects in $(m \downarrow \boldsymbol{CP}[N])$, with $\delta_i = \langle \mu_i, \pi_i, \nu_i \rangle$. Then*

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

Proof. (⇒) Let $\delta_1 \;.\; \delta_2$, 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}, and thus, by Definition 5.23, either $t_2' {}^\frown \iota(t_1)$ or $t_2' < \iota(t_1)$. To conclude that ι is rigid we must show that in both cases t_2' is in O_{π_1}.

- If $t_2' {}^\frown \iota(t_1)$, since the process π_2 is deterministic, t_2' and $\iota(t_1)$ cannot be in conflict and thus it must be $\underline{t_2'} \cap {}^\bullet\iota(t_1) \neq \emptyset$. Since t_2' uses as context a place which is not maximal in O_{π_1}, necessarily t_2' is in O_{π_1}, otherwise it could not be added by concatenating π to π_1.
- If $t_2' < \iota(t_1)$ then we can find a transition t_3' in O_{π_2} such that $t_2' < t_3'$ and $t_3'^{\bullet} \cap ({}^\bullet\iota(t_1) \cup \underline{\iota(t_1)})$. As above, t_3' must be in O_{π_1} since it uses as postcondition a place in O_{π_1}. An inductive reasoning based on this argument shows that also t_2' is in O_{π_1}.

(⇐) 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 $(O_{\pi_2} \backslash O_{\pi_1}) \cup \max(O_{\pi_1})$, 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. On the other hand 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 $\delta_1 \lesssim \delta_2$. ■

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 9.39 (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 9.3. *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 $I_2 \in \mathsf{Idl}(P_2)$ it 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 right implication is obvious. For the left 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. ■

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 exists 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 9.1 (Unfolding vs. Concatenable Processes). *Let N be a semi-weighted c-net. Then* $\mathsf{Idl}((m \downarrow \boldsymbol{CP}[N]))$ *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 $\mathcal{E}_a(\mathcal{U}_a(N))$ (see Theorem 3.1). By Lemma 9.3, 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 $(m \downarrow \mathbf{CP}[N])$,

$$\delta_1 \precsim \delta_2 \quad \text{iff} \quad \xi(\delta_1) \sqsubseteq \xi(\delta_2).$$

The function ξ can be defined as follows. Let $\delta = \langle \mu, \pi, \nu \rangle$ be a concatenable process in $(m \downarrow \mathbf{CP}[N])$. Since π is 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.

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

56

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.[11] Thus $\pi = f_N \circ \pi_C'$ is a deterministic process of N, and, by the definition of ξ, we immediately get that $\xi(\pi) = \pi_C'(T_{\pi_C'}) = C$.

[11] 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.

- 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 9.2, there exists a left-injection $\iota : \delta_1 \to \delta_2$. 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.

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

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 the second condition of Definition 3.14 is also 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:

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

By properties of occurrence net morphisms (Theorem 5.1 and the fact that O_{π_2} is deterministic), this implies $t'_2 \nearrow \iota(t'_1)$ and thus, since ι is 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. 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 5.3), $\pi'_2(t_2) \nearrow \pi'_2(\iota(t_1))$. 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 $\pi'_2(t_2) \in \xi(\delta_1)$, hence 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 9.2, the existence of the left injection $\iota : \delta_1 \to \delta_2$, implies $\delta_1 \lesssim \delta_2$. ■

10. CONTEXTUAL NETS WITH MULTISET CONTEXTS

In this section we discuss how the theory developed in this paper can be extended to deal with the more general class of (semi-weighted) contextual nets where the context of a transition is a multiset rather than a simple set. This is a natural choice if we think of transitions as agents which compute some results, i.e., their post-set, starting from some arguments, i.e., their pre-set, which is destroyed, and their context, which is instead accessed in a nondestructive manner. A token in a place s is thus interpreted as an argument of "type" s and hence the multiplicities of pre-set, post-set, and context of transitions have a very clear meaning: a transitions can consume and read several arguments of the same type and, similarly, produce several results of the same type.

DEFINITION 10.40 (mc-net). A *multiset contextual Petri net (mc-net)* is a tuple $N = \langle S, T, F, C, m \rangle$, where S, T, F and m are defined as for c-nets, while $C : T \leftrightarrow S$ is a multirelation, called the *context multirelation*.

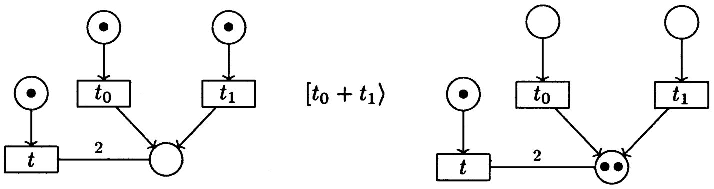


FIG. 13. A simple mc-net.

The *context* of a finite multiset of transitions $A \in \mu T$ is, in this case, a multiset defined as $\underline{\underline{A}} = \mu C(A)$. The notion of *enabling* remains essentially unchanged: a finite multiset of transitions A is enabled by a marking M if, besides the pre-set of A, the multiset M contains 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 not only by many transitions at the same time, but also with multiplicity greater than one by the same transition.

Definition 10.41 (Token Game). Let N be an mc-net and let M be a marking of N. A finite multiset of transitions $A \in \mu T$ is *enabled* by M if ${}^\bullet A + [\![\underline{\underline{A}}]\!] \leq M$. In this case $M\,[A\rangle\, M + A^\bullet - {}^\bullet A$.

Since here we consider contexts with multiplicities, the reader could have expected a notion of enabling requiring for the presence of each context with the corresponding multiplicity, namely

$$M\,[A\rangle \quad \text{iff} \quad {}^\bullet A + \underline{\underline{A}} \leq M. \tag{$\dagger$}$$

We remark that this would not fit with the intuition underlying contextual nets. Consider, for instance, the net N_1 in Fig. 1 and the multiset of transitions $t_0 + t_1$. We have ${}^\bullet(t_0 + t_1) = s_0 + s_1$ and $\underline{\underline{(t_0 + t_1)}} = 2 \cdot s$. According to (†), the marking of N_1 in Fig. 1, namely $s_0 + s_1 + s$ would not enable $t_0 + t_1$, contradicting the idea that a single token in s can be read concurrently by t_0 and t_1.

Still, one could think that, although it is natural to allow contexts to be shared among different transitions, each single transition, to be enabled, should require its context with the right multiplicities. The idea of allowing for the firing of a transition when at least one token is present in each context place can be understood by recalling the interpretation of transitions as agents and of contexts as read-only arguments of such agents: in this view not only different agents can share read-only arguments, but also an agent requiring two "read" parameters of the same type can read twice the same argument. At a more formal level, we have been influenced also by the correspondence between contextual nets and graph transformation systems [17, 20]. In fact, in a graph transformation system, which can be thought of as a "generalized" contextual net, a graph production may specify a context with multiple occurrences of the same resource and can be applied with a match which is noninjective on the context.

According to the multiplicities of places in the context of a transition t, the firing of t may involve a multiset of tokens larger than $[\![\underline{\underline{t}}]\!]$ (ranging from $[\![\underline{\underline{t}}]\!]$ to $\underline{\underline{t}}$). For example, in the net of Fig. 13, after the firing of $t_1 + t_0$, we may have three "different" firings of t, since t can use as context

- both the tokens generated by t_0 and by t_1;
- twice the token generated by t_0;
- twice the token generated by t_1.

In the first case the occurrence of t causally depends both on t_0 and on t_1, in the second case it depends only on t_0, and in the third case only on t_1. More precisely, as the functions ${}^\bullet(.)$, $(.)^\bullet : \mu T \to \mu S$ associate to each multiset of transitions A the multiset of tokens which are consumed and produced by the firing of A, in the presence of contexts we can introduce a relation $\mathsf{read} \subseteq \mu T \times \mu S$ such that $A\,\mathsf{read}\,M$ means that M can be used as context in the firing of A. According to the discussion above, read can be formally defined as: for all finite multisets $A \in \mu T$ and for all $X \in \mu S$,

$$A\,\mathsf{read}\,X \quad \text{iff} \quad [\![\underline{\underline{A}}]\!] \leq X \leq \underline{\underline{A}}.$$

Observe that, different from ${}^\bullet(.)$ and $(.)^\bullet$, which are functions, read is a relation.

A mc-net morphism is still required to preserve the initial marking as well as the pre- and post-sets of transitions, while contexts are preserved in a weak sense.

Definition 10.42 (mc-net Morphism). Let N_0 and N_1 be mc-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 $h_S : S_0 \leftrightarrow S_1$ is a *finitary multirelation* such that

1. $\mu h_S(m_0)$ is defined and $\mu h_S(m_0) = m_1$;
2. for each transition $t \in \mu T_0$, $\mu h_S({}^\bullet t)$, $\mu h_S(t^\bullet)$ and $\mu h_S(\underline{t})$ are defined, and
   (i) $\mu h_S({}^\bullet t) = {}^\bullet \mu h_T(t)$;
   (ii) $\mu h_S(t^\bullet) = \mu h_T(t)^\bullet$;
   (iii) $\llbracket \underline{\underline{\mu h_T(t)}} \rrbracket \leq \mu h_S(\underline{t}) \leq \underline{\underline{\mu h_T(t)}}$.

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

Conditions (1), (2.i), and (2.ii) are the same as in Definition 4.20, but condition (2.iii), regarding contexts, deserves some comments. Like the image of the pre-set (post-set) of t is required to be a multiset of tokens which is the pre-set (post-set) of the image of t, similarly, given a multiset of tokens X which can be used as context by t, its image must be a set of tokens that can be used as context by the image of t. By using the “ read ” notation defined before, this requirement can be expressed as follows: for any $X \in \mu S_0$

$$t \; \mathsf{read} \; X \Rightarrow \mu h_T(t) \; \mathsf{read} \; \mu h_S(X).$$

According to the definition of read, this condition can be rephrased by asking that for any $X \in \mu S_0$, if $\llbracket \underline{t} \rrbracket \leq X \leq \underline{t}$ then $\llbracket \underline{\underline{\mu h_T(t)}} \rrbracket \leq \mu h_S(X) \leq \underline{\underline{\mu h_T(t)}}$, which is in turn equivalent to condition (2.iii) above. It is easy to prove that the token game and thus reachable markings are preserved by mc-net morphisms.

Observe that **CN** is a full subcategory of **MCN**. In fact if N is a c-net, namely an mc-net where the context multirelation C is a relation (i.e., $C = \llbracket C \rrbracket$), then for any transition t, we have $\underline{t} = \underline{t}$. Therefore, when N_0 and N_1 are c-nets, condition (2.iii) in the definition of mc-net morphism above reduces to $\underline{\underline{\mu h_T(t)}} = \mu h_S(\underline{t})$, i.e., to condition (2.iii) in the definition of c-net morphism (Definition 4.20).

If we denote by **SW-MCN** the full subcategory of **MCN** having semi-weighted mc-nets as objects, then the whole theory developed in this paper for **SW-CN**, comprising the coreflective semantics of semi-weighted nets, their process semantics, and the relationship between the two approaches, smoothly extends to the wider category **SW-MCN**. The notion of safe net, occurrence net, and the corresponding categories remains the same. In proving that **O-CN** coreflects in **SW-MCN** we only need to modify the definition of the unfolding (see Definition 6.28). The equation defining the set transitions of the unfolding slightly changes in order to generate a different occurrence of a transition t for each possible multiset of tokens that t can use in its firing:

$$\begin{aligned} T' = \{\langle M_p, M_c, t \rangle \mid \; & M_p, M_c \subseteq S' \;\wedge\; M_p \cap M_c = \emptyset \;\wedge\; conc(M_p \cup M_c) \;\wedge \\ & t \in T \;\wedge\; \mu f_S(M_p) = {}^\bullet t \;\wedge\; \llbracket \underline{t} \rrbracket \leq \mu f_S(M_c) \leq \underline{t}\}. \end{aligned}$$

11. 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:

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

Such a construction is a consistent extension of Winskel’s one [10], 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. More precisely, we have shown how each of our coreflections cuts down to Winskel's coreflection between the corresponding subcategories.

We have also shown that a close relationship exists between the unfolding semantics and the deterministic process semantics, generalizing a result of [9] 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 role 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, such as PES's with possible events [43], flow event structures with possible flow [43], and extended bundle event structures [33]. 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.

We already mentioned that the McMillan algorithm for the construction of a finite prefix of the unfolding has been generalized in [5] to a subclass of safe contextual nets, called read-persistent contextual nets, and it has been applied to the analysis of asynchronous circuits. We are confident that the results in the present paper, and in particular the notion of a set of possible histories of an event in a contextual net, may ease the extension of the technique proposed in [5] from the subclass of read-persistent nets to the whole class of semi-weighted c-nets (perhaps at the price of a growth of the complexity).

Recall that Winskel's construction has been generalized in [9] 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 believe 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 [9] 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 [43], 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 [10].

Finally, we remark that one of the motivations of the research on contextual nets is their relationship with *graph transformation systems* (GTS's) [22, 34], 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 specification of 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 [22], 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 Fig. 14. Informally, to apply such a rule to a graph G we must find an occurrence of its 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 role 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

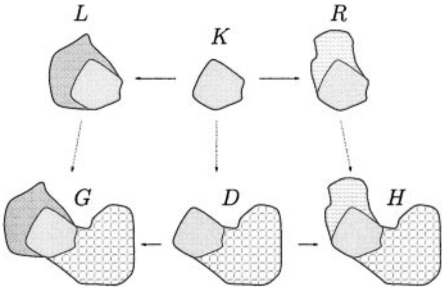


FIG. 14. A (double pushout) graph rewriting step.

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 and distributed systems, people working in this area have been naturally led 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 nontrivial features, makes nontrivial also such generalizations. Some successful results in the project of extending the constructions from net theory to GTS's have been obtained in the development of a theory of nonsequential processes for GTS's [32, 38]. Since contextual nets extend ordinary 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's. Indeed, (a part of) the results of this paper have been recasted for GTS's [12, 30], but a coreflective semantics for GTS's is still missing and constitutes a direction of further research.

ACKNOWLEDGMENTS

We are grateful to the anonymous referees for their insightful comments on the previous version of this paper.

REFERENCES

1. Nielsen, M., Plotkin, G., and Winskel, G. (1981), Petri nets, event structures and domains, Part 1, *Theoret. Comput. Sci.* **13**, 85–108.
2. Pinna, G. M., and Poigné, A. (1992), On the nature of events, *in* "Proceedings of MFCS'92," Lecture Notes in Computer Science, Vol. 629, pp. 430–441, Springer-Verlag, Berlin.
3. Busi, N. (1990), "Petri Nets with Inhibitor and Read Arcs: Semantics, Analysis and Application to Process Calculi," Ph.D. thesis, University of Siena, Department of Computer Science.
4. Degano, P., Meseguer, J., and Montanari, U. (1998), Axiomatizing net computations and processes, *in* "Proceedings of LICS'89," pp. 175–185.
5. Vogler, W., Semenov, A., and Yakovlev, A. (1998), Unfolding and finite prefix for nets with read arcs, *in* "Proceedings of CONCUR'98," Lecture Notes in Computer Science, Vol. 1466, pp. 501–516, Springer-Verlag, Berlin.
6. Busi, N., and Pinna, G. M. (1996), Non sequential semantics for contextual P/T nets, *in* "Application and Theory of Petri Nets" (J. Billington and W. Reisig, Eds.), Lecture Notes in Computer Science, Vol. 1091, pp. 113–132, Springer-Verlag, Berlin.
7. Langerak, R. (1992), Bundle event structures: A non-interleaving semantics for lotos, *in* "5th Intl. Conf. on Formal Description Techniques (FORTE'92)," pp. 331–346, North-Holland, Amsterdam.
8. Vogler, W. (1997), Partial order semantics and read arcs., *in* "Proceedings of MFCS'97" (I. Privara and P. Ruzicka, Eds.), Lecture Notes in Computer Science, Vol. 1295, pp. 508–518, Springer-Verlag, Berlin.
9. Meseguer, J., Montanari U., and Sassone, V. (1996), Process versus unfolding semantics for Place/Transition Petri nets, *Theoret. Comput. Sci.* **153**(1/2), 171–210.
10. Winskel, G. (1987), "Event Structures, Petri Nets: Applications and Relationships to Other Models of Concurrency," Lecture Notes in Computer Science, Vol. 255, pp. 325–392, Springer-Verlag, Berlin.
11. Janicki, R., and Koutny, M. (1995), Semantics of inhibitor nets, *IComp* **123**, 1–16.
12. Baldan, P., Corradini, A., and Montanari, U. (1999), Unfolding and event structure semantics for graph grammars, *in* "Proceedings of FoSSaCS'99" (W. Thomas, Ed.), Lecture Notes in Computer Science, Vol. 1578, pp. 73–89, Springer-Verlag, Berlin.

13. Goguen, J. A. (1991), A categorical manifesto, *in* "Mathematical Structures in Computer Science," Vol. 1, Cambridge Press.
14. Meseguer, J., Montanari, U., and Sassone, V. (1997), On the semantics of Place/Transition Petri nets, *in* "Mathematical Structures in Computer Science," Vol. 7, pp. 359–397, Cambridge Press.
15. Boudol, G., and Castellani, I. (1998), 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," Lecture Notes in Computer Science, Vol. 354, Springer-Verlag, Berlin.
16. Montanari, U., and Rossi, F. (1994), Contextual occurrence nets and concurrent constraint programming, *in* "Proceedings of the Dagstuhl Seminar 9301 on Graph Transformations in Computer Science" (H.-J. Schneider and H. Ehrig, Eds.), Lecture Notes in Computer Science, Vol. 776, Springer-Verlag, Berlin.
17. Corradini, A. (1996), Concurrent graph and term graph rewriting, "Proceedings of CONCUR'96" (U. Montanari and V. Sassone, Eds.), Lecture Notes in Computer Science, Vol. 1119, Springer-Verlag, Berlin.
18. Christensen, S., and Hansen, N. D. (1993), Coloured petri nets extended with place capacities, test arcs and inhibitor arcs., *in* "Applications and Theory of Petri Nets" (M. Ajmone-Marsan, Ed.), Lecture Notes in Computer Science, Vol. 691, Springer-Verlag, Berlin.
19. Baldan, P., Corradini, A., and Montanari, U. (1998), An event structure semantics for P/T contextual nets: Asymmetric event structures, *in* "Proceedings of FoSSaCS '98" (M. Nivat, Ed.), Lecture Notes in Computer Science, Vol. 1378, pp. 63–80, Springer-Verlag, Berlin.
20. Montanari, U., and Rossi, F. (1995), Contextual nets, *Acta Informatica* **32**(6), 545–586.
21. Degano, P., Gorrieri, R., and Vigna, S. (1993), On relating some models for concurrency, *in* "4th Conference on Theory and Practice of Software Development" (M. C. Gaudel and J. P Jouannaud, Eds.), Lecture Notes in Computer Science, Vol. 668, pp. 15–30, Springer-Verlag, Berlin.
22. Ehrig, H. (1987), Tutorial introduction to the algebraic approach of graph-grammars, *in* "Proceedings of the 3rd International Workshop on Graph-Grammars and Their Application to Computer Science" (H. Ehrig, M. Nagl, G. Rosenberg, and A. Rosenfeld, Eds.), Lecture Notes in Computer Science, Vol. 291, pp. 3–14, Springer-Verlag, Berlin.
23. Montanari, U., and Rossi, F. (1993), "Contextual Nets," Technical Report TR-4/93, Dipartimento di Informatica-Università di Pisa.
24. Janicki, R., and Koutny, M. (1991), Invariant semantics of nets with inhibitor arcs, *in* "Proceedings of CONCUR '91," Lecture Notes in Computer Science, Vol. 527, Springer-Verlag, Berlin.
25. Petri, C. A. (1962), "Kommunikation mit Automaten," Ph.D. thesis, Schriften des Institutes für Instrumentelle Matematik, Bonn.
26. Ristori, G. (1994), "Modelling Systems with Shared Resources via Petri Nets," TD-5/94, Department of Computer Science, University of Pisa.
27. Vogler, W. (1996), "Efficiency of Asynchronous Systems and Read Arcs in Petri Nets," Technical Report, Institüt für Mathematik, Augsburg University, 352.
28. Berry, G. (1978), Stable models of typed lambda-calculi, *in* "Proceedings of the 5th ICALP," Lecture Notes in Computer Science, Vol. 62, pp.72–89, Springer-Verlag, Berlin.
29. Gadducci, F., and Montanari, U. (1998), Axioms for contextual net processes, *in* "Proceedings of ICALP'98," Lecture Notes in Computer Science, Vol. 1443, pp. 296–308, Springer-Verlag, Berlin.
30. Baldan, P. (2000), "Modelling Concurrent Computations: From Contextual Petri Nets to Graph Grammars," Ph.D. thesis, Technical Report TD-1/100, Department of Computer Science, University of Pisa.
31. Meseguer, J., Montanari, U., and Sassone, V. (1992), On the semantics of Petri nets, *in* "Proceedings of CONCUR '92," Lecture Notes in Computer Science, Vol. 630, pp. 286–301, Springer-Verlag, Berlin.
32. Corradini, A., Montanari, U., and Rossi, F. (1996), Graph processes, *Fundamenta Informaticae* **26**, 241–265.
33. Langerak, R. (1992), "Transformation and Semantics for LOTOS," Ph.D. thesis, Department of Computer Science, University of Twente.
34. Corradini, A., Montanari, U., Rossi, F., Ehrig, H., Heckel, R., and Löwe, M. (1997), Algebraic approaches to graph transformation I: Basic concepts and double pushout approach, *in* "World Scientific Handbook of Graph Grammars and Computing by Graph Transformation, Vol 1, Foundations" (G. Rozenberg, Ed.), World Scientific, Singapore.
35. McMillan, K. L. (1993), "Symbolic Model Checking" Kluwer, Dordrecht/Norwell, MA.
36. Busi, N., and Gorrieri, R. (1995), A petri nets semantics for π-calculus, *in* "Proceedings of CONCUR'95," Lecture Notes in Computer Science, Vol. 962, pp. 145–159, Springer-Varlag, Berlin.
37. Janicki, R., and Koutny, M. (1993), Structure of concurrency, *Theoret. Comput. Sci.* **112**, 5–52.
38. Baldan, P., Corradini, A., and Montanari, U. (1998), Concatenable graph processes: relating processes and derivation traces, *in* "Proceedings of ICALP'98," Lecture Notes in Computer Science, Vol. 1443, pp. 283–295, Springer-Verlag, Berlin.
39. Winkowsky, J. (1998), "Processes of contextual nets and their characteristics," Technical Report, Institute of Computer Science, Polish Academy of Sciences 847.
40. Boudol, G. (1990), Flow event structures and flow nets, *in* "Semantics of System of Concurrent Processes," Lecture Notes in Computer Science, Vol. 469, pp. 62–95, Springer-Verlag, Berlin.
41. Reisig, W. (1985), "Petri Nets: An Introduction, EACTS Monographs on Theoretical Computer Science," Springer-Verlag, Berlin.
42. Baldan, P., Corradini, A., and Montanari, U. (1999), Unfolding of double-pushout graph grammars is a coreflection, *in* "TAGT'98 Conference Proceedings" (G. Ehrig, G. Engels, H. J. Kreowsky, and G. Rozemberg, Eds.), Lecture Notes in Computer Science, Vol. 1764, pp. 145–163, Springer-Verlag, Berlin.
43. Pinna, G. M., and Poigné, A. (1995), On the nature of events: another perspective in concurrency, *Theoret. Comput. Sci.* **138**, 425–454.