This is the Author's accepted manuscript version of the following contribution:

H. Melgratti, C. A. Mezzina and G. Michele Pinna, "A distributed operational view of Reversible Prime Event Structures," *2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, Rome, Italy, 2021, pp. 1-13, doi: https://doi.org/10.1109/LICS52264.2021.9470623.

When citing, please refer to the published version.

A distributed operational view of Reversible Prime Event Structures

Hernán Melgratti
ICC - Universidad de Buenos Aires, Conicet, Argentina

Claudio Antares Mezzina
Dipartimento di Scienze Pure e Applicate, Università di Urbino, Italy

G. Michele Pinna
Dipartimento di Matematica e Informatica, Università di Cagliari, Italy

Abstract—*Reversible* prime event structures extend the well-known model of prime event structures to represent *reversible* computational processes. Essentially, they give abstract descriptions of processes capable of undoing computation steps. Since their introduction, event structures have played a pivotal role in connecting operational models (traditionally, Petri nets and process calculi) with denotational ones (algebraic domains). For this reason, there has been a lot of interest in linking different classes of operational models with different kinds of event structures. Hence, it is natural to ask which is the operational counterpart of reversible prime event structures. Such question has been previously addressed for a subclass of reversible prime event structures in which the interplay between causality and reversibility is restricted to the so-called *cause-respecting reversible structures*. In this paper, we present an operational characterisation of the full-fledged model and show that reversible prime event structures correspond to a subclass of contextual Petri nets, called *reversible causal nets*. The distinctive feature of reversible causal nets is that causality is recovered from inhibitor arcs instead of the usual overlap between post and presets of transitions. In this way, we are able to operationally explain also out-of-causal order reversibility.

I. INTRODUCTION

Event structures are a well-established model of concurrency. They were originally proposed by Nielsen, Plotkin and Winskel [18] as an intermediate abstraction in between Scott domains (i.e., a denotational model) and Petri nets (i.e., an operational model). While Petri nets describe the behaviour of a system in terms of the consumption and production of data items (i.e., tokens) from repositories (i.e., places), an event structure consists of a set of event occurrences and constraints that regulate such occurrences (i.e., relations over events). Consider the Petri net N depicted in Figure 1a, which consists of five places s_1, s_2, s_3, s_4 and s_5, three transitions a, b and c and two tokens (depicted as bullets) respectively placed on s_1 and s_3. The edges connecting places to transitions describe the consumption and production of tokens; e.g., the firing (i.e., execution) of b consumes a token from each s_2 and s_3 and produces a token in s_4. The tokens available in N enable the firing of the transitions a and c but not that of b because there is no token in s_2. That missing token is produced by the firing of a; hence, b can be fired only after a is so. For this reason, we
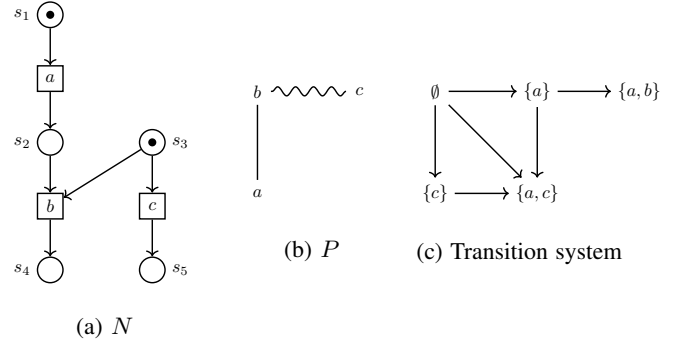
Fig. 1: A simple Petri net and its associated Event Structure

say that b *causally depends* on a. However, b can be fired only if c is not (and vice versa) because each transition requires a token from s_3, which just contains one. In this case, we say that b and c are in *conflict*. An abstract description of the behaviour of N can be given in terms of an event structure, which consists of the definition of the causal dependencies and conflicts as binary relations over a set of events (each event represents a firing of a transition). A graphical representation is shown in Figure 1b, where causality ($<$) is drawn with straight lines (to be read from bottom to top) and binary conflicts ($\#$) are represented by curly lines. In this case, b causally depends on a (i.e., $a < b$), while b and c are in conflict (i.e., $b \# c$).

The behaviour associated with an event structure is understood in terms of a transition system defined over *configurations* (i.e., sets of events), as illustrated in Figure 1c. For instance, the transition $\emptyset \to \{a, c\}$ indicates that the initial state $\emptyset$ (i.e., no event has been executed yet) may evolve to the state $\{a, c\}$ by concurrently executing a and c. Note that neither $\{b\}$ nor $\{a, b, c\}$ are configurations because, on the one hand, b cannot occur without a and, on the other hand, b and c cannot occur in the same run of the system.

Since the seminal work by Winskel [27] that shows a tight connection (via a chain of correflections) between (the category of) safe nets and (prime) event structures, a lot of effort has been made to associate different guises of Petri nets with the corresponding class of event structures (e.g., [4], [12], [6], [26] to name a few). Recently, event structures have been extended to account for *reversible* concurrent systems, namely a class of systems that have lately received lot of attention because of their applications in different fields [1], [16],
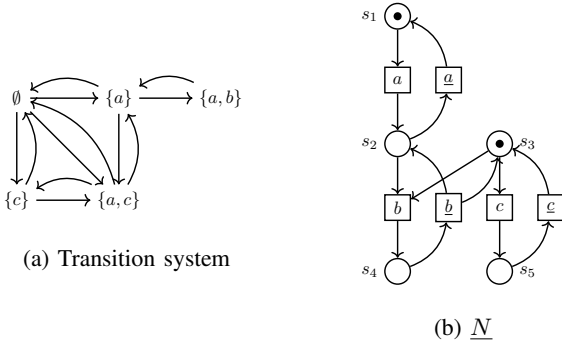
(a) Transition system

(b) $\underline{N}$

Fig. 2: Causal-consistent reversible version of N

(a) $c \triangleright \underline{a}$

(b) $c \prec \underline{a}$

Fig. 3: Prevention and reverse causality operationally

including programming abstractions for reliable systems [8], [13], [10], program analysis and debugging [11], modelling bio-chemical simulations [9] and quantum computing [19]. The distinctive feature of a reversible system is that the execution of actions is liable to be undone. *Reversible prime event structures* (rPES) [21] accommodate the undoing of executed actions by allowing configurations to evolve by removing events. For instance, if c were an undoable event of the event structure P in Figure 1b, then the associated transition system would include the transition $\{a, c\} \to \{a\}$. This is a disruptive feature in event structures since it breaks the underlying assumption by which configurations evolve by adding events. In fact, if X and Y are two configurations of an rPES then $X \to Y$ does not imply $X \subseteq Y$. As a consequence, the existing approaches to recover Petri nets out of event structures, even the most general ones [25], are not applicable. As a matter of fact, we still lack a procedure to associate a Petri net to a given rPES. Previous attempts [15] do this job just for the subclass of *cause-respecting* rPESes, i.e., rPESes that allow the reversing of an event once all events it caused have been reversed. For instance, the transition system associated with a cause-respecting reversible version of the event structure P in Figure 1b is depicted in Figure 2a. Note that each transition $X \to Y$ is paired with a reversing one $Y \to X$; consequently, the configuration $\{a, b\}$ can be reversed only by undoing first b and then a, i.e., $\{a, b\} \to \{a\} \to \emptyset$. Contrastingly, the transition $\{a, b\} \to \{b\}$ is not included because it accounts for the reversal of a before the reversal of the event b, which causally depends on a.

As shown in [15], the transition system of a *cause-respecting* rPES can be implemented (concurrently / distributedly) as a Petri net where the undoing of events is achieved via *reversing* transitions, i.e., each transition t (corresponding to some event of the rPES) is accompanied by another transition $\underline{t}$ that undoes the effects of the firing of t, i.e., $\underline{t}$ (i) consumes the tokens produced by t; and (ii) produces the tokens consumed by t. The transition system in Figure 2a is implemented by the net $\underline{N}$ in Figure 2b, which is essentially the extension of N (Figure 1a) with the reversing transitions $\underline{a}$, $\underline{b}$ and $\underline{c}$.

This approach however falls short when addressing the full expressivity of rPESes, which accommodates different flavours of reversibility [1], [16]. The reversing mechanism of an rPES
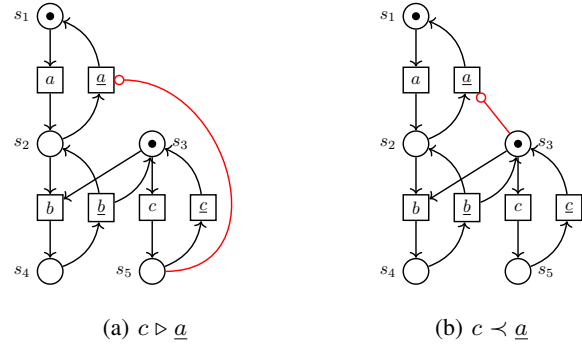
is defined in terms of two relations (additional to the classical causality and conflicts): *prevention* and *reverse causality*. For instance, an rPES can stipulate that some event can be undone only when some other events have not occurred (prevention). For instance, if we stipulate that c prevents the undoing of a (written $c \triangleright \underline{a}$), then $\{a, c\} \to \{c\}$ is banned from the transition system even though a is undoable and c does not causally depend on a. We can also specify that a particular event can be undone only when some other events have already occurred (reverse causality). For instance, if c is a reverse cause of a (written $c \prec \underline{a}$) then a cannot be reversed until c occurs, i.e., the transition $\{a\} \to \emptyset$ is not admissible. We note that these constraints can be translated into Petri nets in the form of contextual arcs; in particular, inhibitor arcs [17], [2] that prevent the firing of a transition if a token is present in some place of the net. For instance, the prevention $c \triangleright \underline{a}$ can be represented in a Petri net with an inhibitor arc in $\underline{a}$, as shown in Figure 3a; the added arc (depicted as $\multimap$) forbids the firing of $\underline{a}$ when s_5 contains a token. Note that s_5 contains a token only when c has been fired, hence $\underline{a}$ cannot be fired if c has occurred. Along the same lines, the reverse causality $c \prec \underline{a}$ can be represented as shown in Figure 3b: in this case, the inhibitor arc is connected to s_3, which will contain a token if c has not been fired. Hence, $\underline{a}$ will be enabled only after c is fired.

Unfortunately, the previous observation is insufficient for capturing the full spectrum of rPESes due to the interplay among causality, prevention and reverse causality. This becomes clear when addressing rPESes enjoying *out-of-causal* order reversibility, which is typical in bio-chemical reactions [22]. Consider again the reversible system in Figure 1b. Assume now that a can be undone also in an out-of-causal order fashion, i.e., a can be reversed independently of the events that it may have caused (which in this case is b). Hence, the transition system would be extended to include the transition $\{a, b\} \to \{b\}$, in which a is reversed even though b is not, and also $\{b\} \to \{b, a\}$, in which the minimal event a is executed. When looking at the net $\underline{N}$ in Figure 2b, the transition $\{a, b\} \to \{b\}$ would require to be able to fire the reversing transition $\underline{a}$ also when the place s_2 does not contain any token (because the firing of b has consumed that token). Hence, a more involved definition of $\underline{a}$ would be needed for
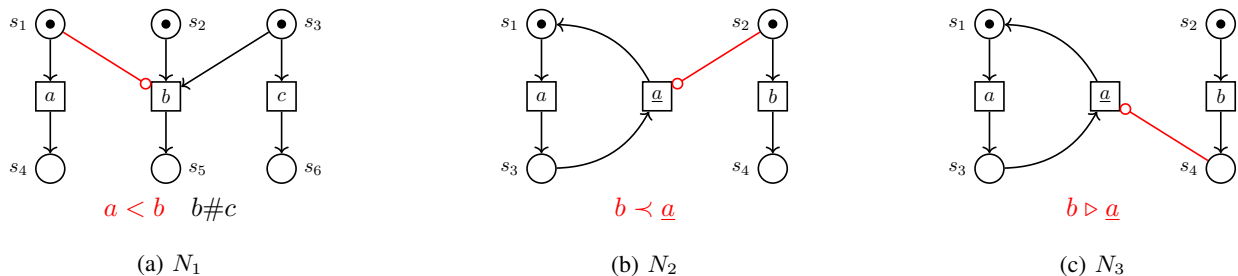
Fig. 4: Examples on how inhibitor acts (—○ arcs) can be used to model causality $<$, reverse causality $\prec$ and prevention $\rhd$ relation among events

handling the undoing of a. Moreover, after reversing a we should be able to fire a again, since the transition system associated to the rPES allows both $\{a\} \to \emptyset \to \{a\}$ and $\{a, b\} \to \{b\} \to \{a, b\}$. It should be noted that the execution of a has different effects in the two computations above: while the configuration $\{a\}$ allows for the firing of b, the configuration $\{a, b\}$ does not (because, b has been already fired). The way in which the causality relation between a and b is described in $\underline{N}$ (Figure 2b) would be insufficient to distinguish the cases above.

In this paper, we take a different approach by observing that inhibitor arcs can be used to model also causality. This simple idea is rendered by the net in Figure 4a, which is an operational counterpart of the event structure P in Figure 1b. The inhibitor arc in Figure 4a is used to model causality among the events a and b. Indeed, b can happen only after a has happened, hence $a < b$. As previously discussed, we represent prevention and reverse causality with inhibitor arcs, as illustrated in Figures 4b and 4c. On the one hand, the inhibitor arc in Figure 4b models the reverse causality $b \prec \underline{a}$, i.e., a can be reversed only when b has been executed. On the other hand, the inhibitor arc in Figure 4c models prevention $b \rhd \underline{a}$, i.e., the undoing of a cannot be executed if the event b has happened. A liberal usage of inhibitor arcs would not do the work. Therefore, we impose some (structural) constraints on nets to achieve our purpose, namely to identify a subclass of nets with inhibitor arcs that corresponds to reversible prime structures. The main contribution of our work is the definition of an operational interpretation of rPESes in terms of (a proper subclass of) Petri nets with inhibitor arcs, called *reversible causal nets*.

This paper is structured as follows: we start by recalling the basics of prime event structures and reversible prime event structures (Section II), and those of nets with inhibitor arcs (Section III). Then, we introduce *causal nets*, which are subclass of nets with inhibitor arcs, and show that they are a suitable counterpart of prime event structures (Section IV). In Section V, we introduce a *reversible* notion of causal nets and prove that they fully correspond to reversible prime event structures, thus giving a proper operational model of them. In Section VI we compare and discuss the relationship between a more classical operational model for prime event structures, namely occurrence nets, with our proposal, and illustrates the reasons that undermine the possibility of generalising

occurrence nets to cope with richer models of reversibility as, e.g., out-of-causal order reversibility.

II. EVENT STRUCTURES

In this section we summarise the basics of *prime* event structures and *reversible* prime event structures by following the presentation in [21]. In what follows, we say that a binary relation $< \subseteq A \times A$ is an *irreflexive partial order* whenever it is irreflexive and transitive, and use $\leq$ to denote its reflexive closure. We write $\lfloor a \rfloor_<$ for the set $\{a' \in A \mid a' \leq a\}$ and shall omit the subscript $<$ when it is clear from the context.

A. pre-PES and PES

Pre-*prime event structures* are a relaxed form of prime event structures in which conflict heredity may not hold. They play a key role in the definition of reversible prime event structures.

Definition 1: A *pre-prime event structure* (pPES) is a triple $P = (E, <, \#)$, where

- E is a countable set of *events*;
- $\# \subseteq E \times E$ is an irreflexive and symmetric relation, called the *conflict relation*; and
- $< \subseteq E \times E$ is an irreflexive partial order, called the *causality relation*, defined such that $\forall e \in E. \ \lfloor e \rfloor_<$ is finite and $\forall e', e'' \in \lfloor e \rfloor_<. \ \neg(e' \# e'')$.

We say a pPES P is a *prime event structure* (PES) when $\#$ is *hereditary* with respect to $<$, i.e., if $e \# e' < e''$ then $e \# e''$ for all $e, e', e'' \in E$.

We write $\mathsf{CF}(X)$ when $X \subseteq E$ is a *conflict-free* set of events, i.e., for all $e, e' \in X$ it holds that $\neg(e \# e')$. Note that for all $Y \subseteq X \subseteq E$, $\mathsf{CF}(X)$ implies $\mathsf{CF}(Y)$.

Example 1: Let $P = (E, <, \#)$ be defined such that

$$E = \{a, b, c, d\} \qquad \# = \{(a, b), (b, a)\} \qquad < = \{(b, c)\}$$

It is immediate to check that $\#$ is irreflexive and symmetric, and $<$ is an irreflexive partial order. Moreover, if $e \in E$ and $e \neq c$ then $\lfloor e \rfloor = \{e\}$, which is finite and conflict free. Additionally, $\lfloor c \rfloor = \{b, c\}$, which is also finite and conflict free. Hence, P is a pPES. However, P is not a PES because conflicts are not inherited along $<$ because $a\#b < c$ but $a\#c$ does not hold. P would be a PES if $\#$ were defined as $\{(a, b), (b, a), (a, c), (c, a)\}$.

The following definition introduces the notion of (labelled) transitions between sets of events of a pPES.

Definition 2: Let $P = (E, <, \#)$ be a pPES and $X \subseteq E$ a conflict-free set of events. We say $A \subseteq E$ is *enabled* at X if

- $A \cap X = \emptyset$ and $\mathsf{CF}(X \cup A)$, and
- $\forall e \in A$. if $e' < e$ then $e' \in X$.

If A is enabled at X, then $X \xrightarrow{A} Y$ where $Y = X \cup A$.

Example 2: Consider the pPES P in Example 1. The sets $\{a\}$, $\{b\}$, $\{d\}$, $\{a, d\}$, and $\{b, d\}$ are all enabled at $\emptyset$ because they are all conflict free and they contain just minimal elements (according to $<$). Then, we can derive, e.g., the transitions $\emptyset \xrightarrow{\{a\}} \{a\}$ and $\emptyset \xrightarrow{\{b,d\}} \{b, d\}$. On the contrary, neither $\{a, b\}$ nor $\{c\}$ are enabled at $\emptyset$. The former, because a and b are in conflict; the latter because $\emptyset$ does not contain b, which is a cause of c. Moreover, $\{c\}$ is enabled at $\{b\}$, because the unique cause of c is b; consequently, $\{b\} \xrightarrow{\{c\}} \{b, c\}$ holds.

The notion of configurations of a pPES is formally defined below.

Definition 3: Let $P = (E, <, \#)$ be a pPES. A set of events $X \subseteq E$ is a *reachable configuration* if it is conflict free, i.e., $\mathsf{CF}(X)$, and there exists a sequence $A_1, \ldots, A_n$, such that $X_i \xrightarrow{A_i} X_{i+1}$ for all i, $X_1 = \emptyset$ and $X_{n+1} = X$. We write $\mathsf{Conf}_{pPES}(P)$ for the set of all reachable configurations of P.

We shall write $\mathsf{Conf}_{PES}(P)$ instead of $\mathsf{Conf}_{pPES}(P)$ when P is a PES.

Example 3: Consider again the pPES P in Example 1 and note that some conflict-free sets of events do not correspond to a state of the computation of P. For instance, the set $\{c, d\}$, which is conflict free, is not a configuration of P because it cannot be reached from the initial state $\emptyset$: c cannot be introduced to a configuration that does not contain b.

The following definition and result from [21] highlight that we can always recover a PES out of a pPES by making conflicts hereditary.

Definition 4: Let $P = (E, <, \#)$ be a pPES. Then $\mathsf{hc}(P) = (E, <, \sharp)$ is the *hereditary closure* of P, where $\sharp$ is derived by using the following rules

$$\frac{e \# e'}{e \sharp e'} \qquad \frac{e \sharp e' \quad e' < e''}{e \sharp e''} \qquad \frac{e' \sharp e}{e \sharp e'}$$

Proposition 1: Let $P = (E, <, \#)$ be a pPES, then

- $\mathsf{hc}(P) = (E, \leq, \sharp)$ is a PES,
- if P is a PES, then $\mathsf{hc}(P) = P$, and
- $\mathsf{Conf}_{pPES}(P) = \mathsf{Conf}_{PES}(\mathsf{hc}(P))$.

B. Reversible prime event structures

We now recall the notion of *reversible prime event structure* following the presentation in [21]. Reversible event structures extend PESes by allowing *some* of their events to be *reversible* or *undoable*. A reversible event u (implicitly) has an associated reversing event $\underline{u}$ capable of removing its effects, i.e., the execution of u followed by $\underline{u}$ cannot be observed. For this reason, the configurations of a reversible prime event structure may not evolve monotonically: reversible events can disappear to account for the fact that its reversing event has been executed. In order to account for different flavours of reversibility, a reversible prime event structure is equipped with two relations,

dubbed *prevention* and *reverse causality*, dictating the way in which reversing events can be executed.

Definition 5: A *reversible prime event structure* (rPES) is a tuple $\mathsf{P} = (E, U, <, \#, \prec, \rhd)$ where $(E, <, \#)$ is a pPES, $U \subseteq E$ are the *reversible/undoable* events (with reverse events being denoted by $\underline{U} = \{\underline{u} \mid u \in U\}$ and disjoint from E, i.e., $\underline{U} \cap E = \emptyset$) and

1) $\prec \; \subseteq E \times \underline{U}$ is the *reverse causality* relation and it is such that $u \prec \underline{u}$ for each $u \in U$ and $\{e \in E \mid e \prec \underline{u}\}$ is finite and conflict-free for every $u \in U$,
2) $\rhd \; \subseteq E \times \underline{U}$ is the *prevention* relation defined such that $\prec \cap \rhd = \emptyset$,
3) the *sustained causation* $\ll$ is a transitive relation defined such that if $e \ll e'$ then
   - $e < e'$,
   - if $e \in U$ then $e' \rhd \underline{e}$,
4) $\#$ is hereditary with respect to $\ll$: if $e \# e' \ll e''$, then $e \# e''$.

The reverse causality relation prescribes the events that are required for the execution of each reversing event, i.e., $e \prec \underline{u}$ says that $\underline{u}$ can be executed (or equivalently, u can be undone) only when e is present. Hence, the condition $u \prec \underline{u}$ stands for the fact that an event u can be undone only when it is present in a configuration. The prevention relation models instead the cases in which an event can be reversed only if some other event is not in the configuration: $e \rhd \underline{u}$ means that u can be reversed only if e is not in the configuration.

Despite the underlying structure $(E, <, \#)$ is a pPES and, hence, conflicts may not be inherited through causality, the definition requires conflicts to be inherited through the new *sustained causation* relation $\ll$, which is coarser that $<$, and accounts for the fact that the causes of some events may disappear from a configuration (see Example 8).

Example 4: Let $\mathsf{P}_1 = (E, U, <, \#, \prec, \rhd)$ be an rPES defined such that

$$
\begin{aligned}
E &= \{a, b, c, d\} & U &= \{b, c\} \\
< &= \{(b, c)\} & \# &= \{(a, b), (b, a), (a, c), (c, a)\} \\
\prec &= \{(b, \underline{b}), (c, \underline{c})\} & \rhd &= \{(c, \underline{b})\}
\end{aligned}
$$

While b and c are reversible in P_1, a and d are not because $a, d \notin U$. Moreover, c causally depends on b because $b < c$; and a is in conflict with both b and c because of the definition of $\#$. Consequently, d is concurrent w.r.t. a, b and c. The definition of the reverse causality contains the expected pairs stating that each reversible event can be undone only if it has been executed (i.e., $b \prec \underline{b}$ and $c \prec \underline{c}$). The definition of the prevention relation states that b cannot be reversed if c has been executed, i.e., $c \rhd \underline{b}$, which is typical of causal reversible models. In this example, sustained causation coincides with causality, i.e., $\ll \; = \; <$, since $b \ll c$ holds because $b < c$ and $c \rhd \underline{b}$ do so. Note that conflicts are inherited along the sustained causation (and also causality), i.e., $a \# b < c$ and $a \# c$.

Example 5: Let $\mathsf{P}_2 = (E, U, <, \#, \prec, \rhd)$ be a variant of P_1 above obtained by extending the definition of $\prec$ with the pair $(d, \underline{c})$. In this case, the event c can be reversed only after d has been executed. This is an example of non-causal

reversibility, in which concurrency interferes with reversibility: c can only be reversed if a concurrent (hence, unrelated) event is executed.

The following definition extends the notion of transitions between sets of events to account for reversing events.

Definition 6: Let $\mathsf{P} = (E, U, <, \#, \prec, \rhd)$ be an rPES and $X \subseteq E$ be a set of events such that $\mathsf{CF}(X)$. For $A \subseteq E$ and $B \subseteq U$, we say that $A \cup \underline{B}$ is *enabled* at X if

- $A \cap X = \emptyset$, $B \subseteq X$ and $\mathsf{CF}(X \cup A)$,
- $\forall e \in A, e' \in E.$ if $e' < e$ then $e' \in X \setminus B$,
- $\forall e \in B, e' \in E.$ if $e' \prec \underline{e}$ then $e' \in X \setminus (B \setminus \{e\})$,
- $\forall e \in B, e' \in E.$ if $e' \rhd \underline{e}$ then $e' \notin X \cup A$.

If $A \cup \underline{B}$ is enabled at X then $X \xrightarrow{A \cup \underline{B}} Y$ where $Y = (X \setminus B) \cup A$.

Example 6: Consider the rPES P_1 introduced in Example 4. We have, e.g., $\emptyset \xrightarrow{\{a\}} \{a\} \xrightarrow{\{d\}} \{a, d\}$, and $\emptyset \xrightarrow{\{b,d\}} \{b, d\} \xrightarrow{\{b\}} \{d\}$, and also $\emptyset \xrightarrow{\{b\}} \{b\} \xrightarrow{\{c\}} \{b, c\} \xrightarrow{\{d,c\}} \{b, d\} \xrightarrow{\{c\}} \{b, c, d\}$. Consider now the rPES P_2 in Example 5, and note that the set $\{d, \underline{c}\}$ is not enabled at $\{b, c\}$: despite c is present, the reverse causality $d \prec \underline{c}$ states that d is also required for the reversing of c.

The reachable configurations of an rPES are the sets of events that can be reached from the empty set by performing events or undoing previously performed events, as stated below.

Definition 7: Let $\mathsf{P} = (E, U, <, \#, \prec, \rhd)$ be an rPES and $X \subseteq E$ a conflict-free set of events, i.e., $\mathsf{CF}(X)$ holds. We say that X is a *(reachable) configuration* if there exist two sequences of sets A_i and B_i, for $i = 1, \dots, n$, such that

- $A_i \subseteq E$ and $B_i \subseteq U$ for all i, and
- $X_i \xrightarrow{A_i \cup B_i} X_{i+1}$ for all i with $X_1 = \emptyset$ and $X_{n+1} = X$.

The set of configurations of P is denoted by $\mathsf{Conf}_{r\text{PES}}(\mathsf{P})$. We say that two rPESes P_1 and P_2 are *equivalent*, written $\mathsf{P}_1 \equiv \mathsf{P}_2$, if they have the same set of configurations, i.e., $\mathsf{Conf}_{r\text{PES}}(\mathsf{P}_1) = \mathsf{Conf}_{r\text{PES}}(\mathsf{P}_2)$.

Example 7: Consider the rPES P_1 introduced in Example 4. Its set of configurations $\mathsf{Conf}_{r\text{PES}}(\mathsf{P}_1)$ is

$$\{\emptyset, \{a\}, \{b\}, \{d\}, \{a, d\}, \{b, d\}, \{b, c\}, \{b, c, d\}\}$$

(the corresponding sequences of transitions are shown in Example 6)

A feature of rPESes that is challenging to accommodate in an operational model is the out-of-causal order reversibility, which is illustrated with the example below.

Example 8: Consider $\mathsf{P}_3 = (E, U, <, \#, \prec, \rhd)$ as a variant of the rPES P_2 in Example 5 defined as follows

$$
\begin{aligned}
E &= \{a, b, c, d\} & U &= \{b, c\} \\
< &= \{(b, c)\} & \# &= \{(a, b), (b, a)\} \\
\prec &= \{(b, \underline{b}), (c, \underline{c}), (d, \underline{c})\} & \rhd &= \emptyset
\end{aligned}
$$

In this case, the induced sustained causation is empty, i.e., $b \not\ll c$ because $b < c$ holds but $c \rhd \underline{b}$ does not. Since $c \rhd \underline{b}$ does not hold in P_3, b can be reversed even when c, which causally depends on b, has been executed, e.g.,

$\emptyset \xrightarrow{\{b\}} \{b\} \xrightarrow{\{c\}} \{b, c\} \xrightarrow{\{b\}} \{c\} \xrightarrow{\{a\}} \{a, c\}$. Note that the configuration $\{a, c\}$ would be banned in standard PESes because $a \# b < c$. However, it is allowed in rPESes because conflicts are not necessarily inherited through causality, which is essential to accommodate out-of-causal-order reversibility.

The previous examples highlight the main challenges to address when developing a net semantics for rPESes, which can be summarised as follows:

- configurations may drop events during computation, i.e., $\{b, c\} \xrightarrow{\{b\}} \{c\}$ in P_3;
- reachable configurations do not necessarily contain all the *causes*, e.g., the configuration $\{c\}$ in P_3 does not contain b though $b < c$;
- causes can be re-enabled, e.g., b is enabled at $\{c\}$ in P_3, however the re-execution of a disables b again, i.e., b is not enabled at $\{a, c\}$;
- conflicts are not inherited through causality, as in P_3;
- reversibility induces dependencies on concurrent events, e.g., c in P_2 can be reversed only after the concurrent event d has been executed.

III. Nets

We start by recalling some notions that will be used throughout this paper. $\mathbb{N}$ denotes the set of natural numbers. A *multiset* over a set A is a function $m : A \to \mathbb{N}$. We assume multisets to be equipped with the usual operations of union $(+)$ and difference $(-)$, and write $m \subseteq m'$ if $m(a) \leq m'(a)$ for all $a \in A$. We shall write $[\![m]\!]$ for the underlying set of a multiset m, i.e., the multiset defined such that $[\![m]\!](a) = 1$ if $m(a) > 0$ and $[\![m]\!](a) = 0$ otherwise. We often confuse a multiset m with the set $\{a \in A \mid m(a) \neq 0\}$ when $m = [\![m]\!]$. In such cases, we write $a \in m$ instead of $m(a) \neq 0$, and $m \subseteq A$ if $m(a) = 1$ implies $a \in A$. Furthermore, we will use standard operations on sets, such as $\cap$, $\cup$ or $\setminus$. The set of all multisets over A is denoted by μA. We write 0 for the unique multiset defined such that $[\![0]\!] = \emptyset$.

A. Nets with inhibitor arcs

We summarise the basics of Petri net with inhibitor arcs along the lines of [17], [2].

Definition 8: A *Petri net* is a 4-tuple $N = \langle S, T, F, \mathsf{m} \rangle$ where S is a set of *places*, T is a set of *transitions* such that $S \cap T = \emptyset$, $F \subseteq (S \times T) \cup (T \times S)$ is the *flow* relation, and $\mathsf{m} \in \mu S$ is the *initial marking*.

Definition 9: A *Petri net with inhibitor arcs* (IPT for short) is a tuple $N = \langle S, T, F, I, \mathsf{m} \rangle$, where $\langle S, T, F, \mathsf{m} \rangle$ is a Petri net, and $I \subseteq S \times T$ is the *inhibiting* relation.

Given an IPT $N = \langle S, T, F, I, \mathsf{m} \rangle$ and $x \in S \cup T$, the *pre-* and *postset* of x are respectively defined as the (multi)sets ${}^\bullet x = \{y \mid (y, x) \in F\}$ and $x^\bullet = \{y \mid (x, y) \in F\}$. If $x \in S$ then ${}^\bullet x \in \mu T$ and $x^\bullet \in \mu T$; analogously, if $x \in T$ then ${}^\bullet x \in \mu S$ and $x^\bullet \in \mu S$. The *inhibitor set* of a transition t is the (multi)set ${}^\circ t = \{s \mid (s, t) \in I\}$. The definition of ${}^\bullet \cdot, \cdot^\bullet, {}^\circ \cdot$ generalise straightforwardly to multisets of transitions.

Example 9: Consider the simple IPT N_1 depicted in Figure 4a, which consists of six places depicted as circles

and three transitions drawn as boxes. The flow relation is represented by black arrows while the inhibitor relation is shown by red lines ended with a small circle. The initial marking $m = \{s_1, s_2, s_3\}$ is represented by the bullets drawn within the corresponding places. Consider the transition b, which consumes tokens from s_2 and s_3, produces a token in s_5, and is inhibited by s_1. Hence, its pre-, post- and inhibiting sets are respectively $^\bullet b = \{s_2, s_3\}$, $b^\bullet = \{s_5\}$, and $^\circ b = \{s_1\}$.

A (multiset of) transition(s) $A \in \mu T$ is *enabled at a marking* $m \in \mu S$, written $m\, [A\rangle$, whenever $^\bullet A \subseteq m$ and $\forall s \in [\![^\circ A]\!].\ m(s) = 0\ \wedge\ A^\bullet(s) = 0$. The last condition requires the absence of tokens in all places connected via inhibitor arcs to the transitions in $[\![A]\!]$. Observe that the multiset 0 is enabled at every marking. A (multiset of) transition(s) A enabled at a marking m can *fire* and its firing produces the marking $m' = m - {}^\bullet A + A^\bullet$. The firing of A at a marking m is denoted by $m\, [A\rangle\, m'$. We assume that each transition t of an IPT N is defined such that $^\bullet t \neq \emptyset$, i.e., it cannot fire *spontaneously* in an uncontrolled manner without consuming tokens.

Example 10: Consider the IPT introduced in Example 9 and note that both a and c are enabled at the initial marking m. On the contrary, b is not enabled because its inhibitor place s_1 contains a token. The firing of a produces the marking $m' = \{s_2, s_3, s_4\}$, i.e. $m\, [a\rangle\, m'$, at which b becomes enabled because its preset s_2 and s_3 is marked while its inhibitor place s_1 is not.

A marking m is *reachable* in N if there exists a firing sequence $\sigma = m\, [A_0\rangle\, m_1\, \cdots\, m_n\, [A_n\rangle\, m$ from the initial marking m to m. We write $\mathcal{M}_N$ for the set of all reachable markings of N. A multiset of transitions X is a *state* of N if there exists a firing sequence $\sigma = m_0\, [A_0\rangle\, m_1\, \cdots\, m_{n-1}\, [A_n\rangle\, m_n$ such that $X = \sum_{0 \leq i \leq n} A_i$. We write $\mathsf{St}(N)$ for the set of all states of N. We say that N_1 and N_2 are *equivalent*, written $N_1 \equiv N_2$, if they have the same set of states, i.e., $\mathsf{St}(N_1) = \mathsf{St}(N_2)$.

Example 11: The set of reachable markings of the IPT in Example 9 is $\mathcal{M}_{N_1} = \{m, \{s_2, s_3, s_4\}, \{s_1, s_2, s_6\}, \{s_4, s_5\}, \{s_2, s_4, s_6\}\}$ while it set of states is $\mathsf{St}(N_1) = \{\emptyset, \{a\}, \{c\}, \{a, b\}, \{a, c\}\}$.

An IPT N is *safe* if each reachable marking is a set, i.e., $\forall m \in \mathcal{M}_N.\, m = [\![m]\!]$. Hereafter, we will consider only safe IPT.

Given a net $N = \langle S, T, F, m \rangle$, we write $<_N$ for the transitive closure of F. We say N is *acyclic* if $\leq_N$ is a partial order.

IV. CAUSAL NETS

We now restrict our attention to the representation of causal dependencies through inhibitor arcs: we introduce a class of contextual Petri nets, dubbed *causal nets*, in which causality is recovered from the inhibiting relation instead of the usual flow relation. The main result in this section (Theorem 1) states that causal nets are an adequate operational counterpart of PESes. As a matter of fact, they are tightly connected with occurrence nets (as discussed in Section VI).

We define the relation $\lessdot$ between transitions t, t' of an IPT by $t \lessdot t'$ iff $^\bullet t \cap {}^\circ t' \neq \emptyset$, i.e., the firing of t consumes (at least) one of the tokens that inhibit the firing of t'. Similarly, we define $\natural$ by $t \natural t'$ iff $^\bullet t \cap {}^\bullet t' \neq \emptyset$.

Definition 10: Let $C = \langle S, T, F, I, m \rangle$ be an IPT. C is a *pre-causal* net (pCN) if the following conditions are satisfied:

1) $\lessdot_C \cap (T \times T) = \emptyset$;
2) $[\![T^\bullet]\!] = T^\bullet$;
3) $\forall t \neq t' \in T.\ {}^\bullet t \cap {}^\bullet t' \cap {}^\circ T = \emptyset$,
4) $\forall t \in T.\ {}^\circ t$ is finite;
5) $\lessdot$ is an irreflexive partial order;
6) $\forall t', t'' \in \lfloor t \rfloor_\lessdot.\ t' \natural t'' \Rightarrow\ t' = t''$; and
7) $m = {}^\bullet T$ and $^\circ T \subseteq m$.

We say a pCN C is a *causal* net (CN) if it also satisfies the following condition

$$\forall t, t', t'' \in T.\ t \natural t' \wedge\ t' \lessdot t'' \Rightarrow\ t \natural t''$$

The conditions imposed on CNs share motivations with those posed on occurrence nets, unravel nets [23], [6], [7] and flow nets [5], which are aimed at explaining computations without resorting to firing sequences. The first condition, which can be equivalently written as $\forall t \in T.\ \forall s \in {}^\bullet t.\ {}^\bullet s = \emptyset$, implies that causal dependencies in CNs do not arise because of the flow relation since $t^\bullet \cap {}^\bullet t' = \emptyset$ holds for all $t, t' \in T$. The second condition ensures that there are no backward conflicts, i.e., a place belongs at most to the postset of one transition. The third one states that any place appearing in the preset of at least two transitions cannot be connected via inhibitor arcs to other places of the net. In this way, we ban or-causality, i.e., situations in which the firing of a transition may have different causes. In combination with the fourth requirement, they ensure that each transition has a finite set of causes, i.e., the set $\{t' \in T \mid t' \lessdot t\}$ is finite. The fifth condition avoids cycles in the dependencies arising from inhibitor arcs. Since $\lessdot$ is irreflexive, none of the transitions is blocked (i.e., $^\bullet t \cap {}^\circ t = \emptyset$ for all t). The sixth condition is reminiscent of the requirement of pPES by which inheritance of conflicts holds locally, i.e., any pair of different transitions t', t'' in $\lfloor t \rfloor_\lessdot$ are not in *conflict*. The last requirement states that the preset of all transitions are initially marked (i.e., $m = {}^\bullet T$). Moreover, $^\circ T \subseteq m$ together with the first requirement imply that inhibitor places cannot appear in the postset of any transition.

The additional condition for CNs imposes conflict inheritance along the relation $\lessdot$ (recall that conflicts are modelled through shared input places).

Example 12: The IPT C_1 depicted in Figure 5a is a pCN, as discussed below. Conditions 1 and 2 of Definition 10 hold because every place appearing in the postset of some transition has exactly one incoming and none outgoing arc. The only place that belongs to the preset of at least two transitions is s_2. However, there is no inhibitor arc connected to s_2. Hence, Condition 3 also holds. Condition 4 follows immediately because there is just one inhibitor arc. The induced causality relation is the irreflexive partial order $\lessdot = \{(b, c)\}$, which satisfies Condition 5. If $t \in T$ and $t \neq c$ then $\lfloor t \rfloor_\lessdot = \{t\}$.
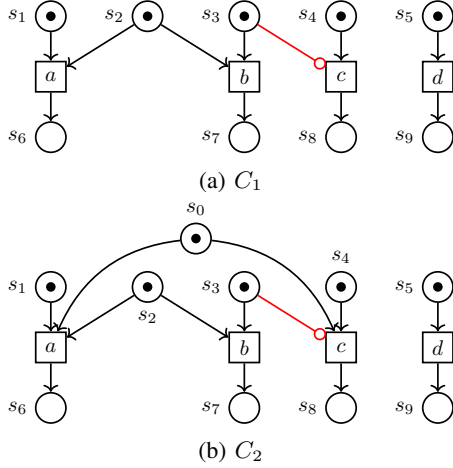
(a) C_1

(b) C_2

Fig. 5: Two pre-causal nets

Fig. 6: $\mathcal{A}(\mathsf{P})$

For $t = c$ we have $\lfloor c \rfloor_< = \{b, c\}$ and ${}^\bullet b$ and ${}^\bullet c$ are disjoint, which satisfies Condition 6. Condition 7 is immediate.

We remark that C_1 is not a CN because conflicts are not inherited along $<$. In fact, $a \natural b$ and $b < c$ but $a \natural c$. The IPT C_2 in Figure 5b, which makes explicit the conflict between a and c, is a CN.

Definition 11: Let $C = \langle S, T, F, I, \mathsf{m} \rangle$ be a pCN. A set of transitions $X \subseteq T$ is a *configuration* of C if:

- $\forall t, t' \in X. t \natural t' \Rightarrow t = t'$ (*conflict freeness*), and
- $\forall t \in X. \lfloor t \rfloor_< \subseteq X$ (*left closedness* with respect to $<$).

The set of configurations of a pCN C is denoted by $\mathsf{Conf}_{p\mathrm{CN}}(C)$.

Should C be a CN, we write $\mathsf{Conf}_{\mathrm{CN}}(C)$ in lieu of $\mathsf{Conf}_{p\mathrm{CN}}(C)$.

Example 13: Consider the pCN C_1 and the CN C_2 in Figure 5. Their sets of configurations coincide and are shown below

$$\mathsf{Conf}_{p\mathrm{CN}}(C_1) = \mathsf{Conf}_{\mathrm{CN}}(C_2) = \{\{a\}, \{b\},$$
$$\{d\}, \{a, d\}, \{b, c\}, \{b, d\}, \{b, c, d\}\}$$

There is a close correspondence between the configurations of a pCN and its reachable markings: any reachable marking determines a configuration of the net and vice versa. This is formally stated by the proposition below.

Proposition 2: Let $C = \langle S, T, F, I, \mathsf{m} \rangle$ be a pCN. Then,

1) if $m' \in \mathcal{M}_C$ then ${}^\bullet m' \in \mathsf{Conf}_{p\mathrm{CN}}(C)$; and
2) if $X \in \mathsf{Conf}_{p\mathrm{CN}}(C)$ then $\mathsf{m} - {}^\bullet X + X^\bullet \in \mathcal{M}_C$.

Example 14: Consider the CN C_2 in Figure 5b. The marking $m' = \{s_1, s_7, s_8, s_9\}$ is reachable because of the following firing sequence:

$$\mathsf{m} \left[\{b, d\}\right\rangle \{s_0, s_1, s_4, s_7, s_9\} \left[\{c\}\right\rangle m'$$

Then, it holds that ${}^\bullet m' = \{b, c, d\}$ is a configuration of C_2 (where $\mathsf{Conf}_{\mathrm{CN}}(C_2)$ is defined in Example 13). Conversely, take $X = \{b, c, d\} \in \mathsf{Conf}_{\mathrm{CN}}(C_2)$, and note that ${}^\bullet X = \{s_0, s_2, s_3, s_4, s_5\}$ and $X^\bullet = \{s_7, s_8, s_9\}$. Then, we have that
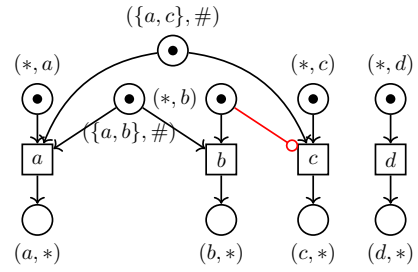
$\mathsf{m} - {}^\bullet X + X^\bullet = \{s_1, s_4, s_7, s_9\} = m'$ is a reachable marking of C_2.

The notion of *equivalence* for causal nets specialises to configurations, i.e., $C_1 \equiv C_2$ iff $\mathsf{Conf}_{p\mathrm{CN}}(C_1) = \mathsf{Conf}_{p\mathrm{CN}}(C_2)$.

A. From pPESes to pCNs

We now show that every pPES can be associated with a pCN that has the same configurations. We start by introducing the mapping $\mathcal{A}$ from pPESes to pCN.

Definition 12: Let $\mathsf{P} = (E, <, \#)$ be a pPES. The associated pCN is $\mathcal{A}(\mathsf{P}) = \langle S, E, F, I, \mathsf{m} \rangle$ where

- $S = \{(*, e) \mid e \in E\} \cup \{(e, *) \mid e \in E\}$
$$\cup \{(\{e, e'\}, \#) \mid e \# e'\},$$
- $F = \{(s, e) \mid s = (*, e) \vee (s = (W, \#) \wedge e \in W)\}$
$$\cup \{(e, s) \mid s = (e, *)\},$$
- $I = \{(s, e) \mid s = (*, e') \wedge e' < e\}$, and
- $\mathsf{m} = \{(*, e) \mid e \in E\} \cup \{(\{e, e'\}, \#) \mid e \# e'\}$.

The construction associates the pPES P with a pCN that has as many transitions as events are in P. Places are identified with pairs, which may have one of the following forms: (i) $(*, e)$ for the precondition of e, $(e, *)$ for the postcondition of e, and $(\{e, e'\}, \#)$ for the conflict $e \# e'$. The flow relation is defined so that each transition e consumes tokens from $(*, e)$ and every $(W, \#)$ where $e \in W$; and only produces a token in $(e, *)$. Different from the classical construction of occurrence nets out of PESes [27], places do not convey causal dependencies, which are modelled here via inhibitor arcs: if e causally depends on e' (i.e., $e' < e$), then there is an inhibitor arc between the transition e and the place $(*, e')$, i.e., a place of the preset of e'. The initial marking m assigns a token to any place appearing in the preset of a transition.

Example 15: Let $\mathsf{P} = (E, <, \#)$ be a pPES where

$$E = \{a, b, c, d\} \qquad < = \{(b, c)\}$$
$$\# = \{(a, b), (b, a), (a, c), (c, a)\}$$

The associated pCN is shown in Figure 6. Note that the transition c is not enabled because of the inhibitor arc from $(*, b)$; all remaining transitions are enabled; also a and b are in conflict because they both consume from $(\{a, b\}, \#)$.

The adequacy of $\mathcal{A}$ is formally stated by showing the equivalence of the respective sets of configurations.

Proposition 3: Let P be a pPES. Then, $\mathcal{A}(\mathsf{P})$ is a pCN and $\mathsf{Conf}_{p\mathrm{PES}}(\mathsf{P}) = \mathsf{Conf}_{p\mathrm{CN}}(\mathcal{A}(\mathsf{P}))$.

Corollary 1: Let P be a PES. Then $\mathcal{A}(\mathsf{P})$ is a CN and $\mathsf{Conf}_{\mathrm{PES}}(\mathsf{P}) = \mathsf{Conf}_{\mathrm{CN}}(\mathcal{A}(\mathsf{P}))$.

B. From pCNs to pPESes

The definition of the mapping $\mathcal{Q}$ from pCNs to pPESes relies on the facts that $\prec$ is an irreflexive partial order, and conflicts are locally preserved in a pCN.

Definition 13: Let $C = \langle S, T, F, I, \mathsf{m} \rangle$ be a pCN. The associated pPES is $\mathcal{Q}(C) = (T, \prec, \#)$ where

$$\# = \{(t, t') \mid t \neq t' \in T \ \wedge \ t \natural t'\}.$$

Example 16: Consider the pCN C_1 of Example 12. Note that, $b \prec c$ holds because $^{\circ}c \cap {}^{\bullet}b \neq \emptyset$ and $a\#b$ (also $b\#a$) because ${}^{\bullet}a \cap {}^{\bullet}b \neq \emptyset$. Hence,

$$\mathcal{Q}(C_1) = (\{a, b, c, d\}, \{(b, c)\}, \{(a, b), (b, a)\}).$$

The PES associated with the CN C_2 of Example 12 is

$$\mathcal{Q}(C_2) = (\{a, b, c, d\}, \{(b, c)\}, \{(a, b), (b, a), (a, c), (b, c)\})$$

which is actually a PES because conflicts are hereditary.

Proposition 4: Let C be a pCN. Then $\mathcal{Q}(C)$ is a pPES and $\mathsf{Conf}_{p\mathrm{PES}}(\mathcal{Q}(C)) = \mathsf{Conf}_{p\mathrm{CN}}(C)$.

Corollary 2: Let C be a CN. Then $\mathcal{Q}(C)$ is a PES and $\mathsf{Conf}_{\mathrm{PES}}(\mathcal{Q}(C)) = \mathsf{Conf}_{\mathrm{CN}}(C)$.

The following result ensures that the notion of causal nets is adequate for PESes. It is obtained by combining Proposition 3 and Corollary 1 (in Section IV-A) with Proposition 4 and Corollary 2 above.

Theorem 1: Let C be a causal net. Then $C \equiv \mathcal{A}(\mathcal{Q}(C))$. Let P be a PES. Then $\mathsf{P} \equiv \mathcal{Q}(\mathcal{A}(\mathsf{P}))$.

V. REVERSIBLE CNS AND REVERSIBLE PESES

In this section we introduce a reversible version of pCNs and show that they are an operational counterpart for rPESes.

A. Reversible causal nets

The intuition behind the definition of a reversible causal net is that of extending a (pre) causal net with transitions, dubbed *backward*, that reverse / undo the effects of previously fired ordinary transitions, which we call *forward*. Given an IPT $N = \langle S, T, F, I, \mathsf{m} \rangle$ and a transition $t \in T$, we write S_t for the set of places $\{s \in S \mid s \in {}^{\bullet}t \ \wedge \ s^{\bullet} = \{t\}\}$, i.e., the places in the preset of t that do not appear in the preset of other transitions.

Definition 14: An IPT $V = \langle S, T, F, I, \mathsf{m} \rangle$ is a *reversible causal net* (rCN) if there exists a partition $\{\overline{T}, \underline{T}\}$ of T, with $\overline{T}$ the forward transitions and $\underline{T}$ the backward ones, such that:

1) $\langle S, \overline{T}, F_{|\overline{T} \times \overline{T}}, I_{|\overline{T} \times \overline{T}}, \mathsf{m} \rangle$ is a pCN net;
2) $[\![{}^{\bullet}\underline{T}]\!] = {}^{\bullet}\underline{T}$ and $\forall \underline{t} \in \underline{T}.\ \exists!\ t \in \overline{T}$ such that $t^{\bullet} = {}^{\bullet}\underline{t}$, ${}^{\bullet}t = \underline{t}^{\bullet}$, and $S_t \cap {}^{\circ}\underline{t} \neq \emptyset$;
3) $\forall \underline{t} \in \underline{T}.\ K = \{t \mid t \in \overline{T} \wedge {}^{\circ}\underline{t} \cap {}^{\bullet}\overline{T} \neq \emptyset\}$ is finite and $[\![{}^{\bullet}K]\!] = {}^{\bullet}K$;
4) $\forall \underline{t} \in \underline{T}.\forall t \in \overline{T}.$ if ${}^{\bullet}t \cap {}^{\circ}\underline{t} \neq \emptyset$ then $t^{\bullet} \cap {}^{\circ}\underline{t} = \emptyset$;
5) $\lll \subseteq \overline{T} \times \overline{T}$ is a transitive relation defined such that $t \lll t'$ if $t \prec t'$ and if there exists $\underline{t} \in \underline{T}$ such that ${}^{\bullet}\underline{t} = t^{\bullet}$ then ${}^{\circ}\underline{t} \cap t'^{\bullet} \neq \emptyset$; and

6) $\forall t, t', t'' \in \overline{T}.\ {}^{\bullet}t \cap {}^{\bullet}t' \neq \emptyset \wedge t' \lll t'' \Rightarrow {}^{\bullet}t \cap {}^{\bullet}t'' \neq \emptyset$.

We write $V^{\underline{T}}$ for an rCN V with backward transitions $\underline{T}$.

The first condition states that the subnet consisting of just forward transitions is a pre causal net. The second one establishes that each backward transition $\underline{t}$ reverses exactly one forward transition t; consequently, $\underline{t}$ consumes the tokens produced by t (i.e., $t^{\bullet} = {}^{\bullet}\underline{t}$) and produces the tokens consumed by t (i.e., $\underline{t}^{\bullet} = {}^{\bullet}t$); the condition $S_t \cap {}^{\circ}\underline{t} \neq \emptyset$ ensures that $\underline{t}$ causally depends on t. By requiring $[\![{}^{\bullet}\underline{T}]\!] = {}^{\bullet}\underline{T}$, we ensure that a forward transition has at most one reversing transition. The remaining conditions recast the ones imposed on rPES, where inhibitor arcs are used to model the reverse causality relation ($\prec$) and the prevention relation ($\rhd$): when an inhibitor arc connects a backward transition with a place in the preset of some forward transition, then the modelled relation is the reverse causality; on the contrary, prevention is represented by linking a backward transition to a place in the the postset of a forward transition. Consequently, the third condition can be read just as requiring each backward transition to causally depend on a finite number of forward transitions (i.e., the ones in K); which moreover should be conflict free (i.e., they do not share places in their preset). By the fourth condition, a backward transition $\underline{t}$ causally dependant on a forward transition t (i.e., ${}^{\bullet}t \cap {}^{\circ}\underline{t} \neq \emptyset$) cannot be prevented by the same transition (i.e., $t^{\bullet} \cap {}^{\circ}\underline{t} = \emptyset$). The relation $\lll$ defined by the fifth condition is analogous to the sustained causation, i.e., it coincides with causality only when prevention enforces causal reversibility. The last condition states that conflicts are inherited along $\lll$.

Example 17: Consider the IPT V in Figure 7, which is an rCN with reversing transitions $\underline{T} = \{\underline{b}, \underline{c}\}$. First note that $\langle S, \overline{T}, F_{|\overline{T} \times \overline{T}}, I_{|\overline{T} \times \overline{T}}, \mathsf{m} \rangle$ is the CN C_2 shown in Figure 5b. There are two backward transitions: $\underline{b}$, for reversing b, and $\underline{c}$ for reversing c. Each of them reverses the markings of its associated forward transition, e.g., $\underline{b}$ consumes from s_7 and produces on s_2 and s_3. The inhibitor arcs connecting s_3 to $\underline{b}$ and s_4 to $\underline{c}$ reflect causality, e.g., $\underline{b}$ causally depends on b. In this case the behaviour of the net would remain unaltered if such inhibitor arcs were omitted. However, we require them in order to uniformly capture causality in terms of $\prec$ instead of resorting to the flow relation. The inhibitor arc connecting s_5 to $\underline{c}$ also accounts for causality: c cannot be fired until d is fired and the token in s_5 is consumed. The inhibitor arc connecting s_8 to $\underline{b}$ models prevention since $\underline{b}$ is not enabled if s_8 is marked (i.e., if c has been fired).

B. From rPESes to rCNes

This section introduces a mapping from rPESes to rCNes. (We prove the adequacy of its definition in Section V-D.)

Definition 15: Let $\mathsf{P} = (E, U, <, \#, \prec, \rhd)$ be an rPES. Then, $\mathcal{A}_r(\mathsf{P}) = \langle S, T, F, I, \mathsf{m} \rangle$ where $\mathcal{A}(E, <, \#) = \langle S, T', F', I', \mathsf{m} \rangle$ and

1) $T = T' \cup \{(u, \mathtt{r}) \mid u \in U\}$;
2) $F = F' \cup \{(s, (u, \mathtt{r})) \mid s \in u^{\bullet}\} \cup \{((u, \mathtt{r}), s) \mid s \in {}^{\bullet}u\}$;
3) $I = I' \cup \{((*, e), (u, \mathtt{r})) \mid e \prec \underline{u}\}$
$$\cup \{((e, *), (u, \mathtt{r})) \mid e \rhd \underline{u}\}.$$

Fig. 7: A simple rCN

Fig. 9: $\mathcal{A}_r(\mathsf{P_3})$
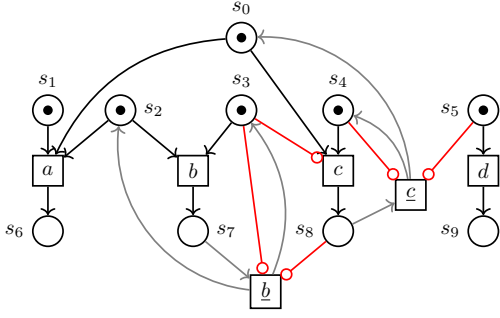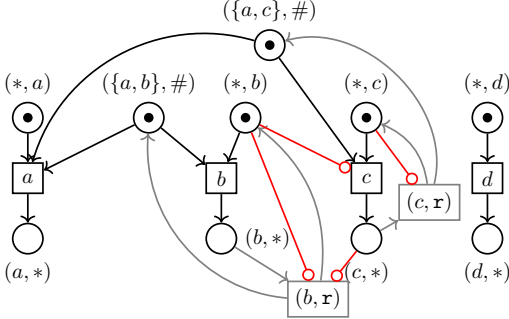

Fig. 8: $\mathcal{A}_r(\mathsf{P_1})$

The construction uses the mapping $\mathcal{A}(\_)$ for obtaining a pCN out of the underlying pPES consisting of just forward events, which is then extended with as many backward transitions as reversible events in P. A backward transition $(u, \mathbf{r})$, which reverses u, is defined such that its preset is the postset of u, and its poset is the preset of u. Inhibitors arcs are as expected: the reverse causality $e \prec \underline{u}$ translates into an arc connecting the backward transition $(u, \mathbf{r})$ to the enabling condition of e, i.e., $(*, e)$; the prevention $e \triangleright \underline{u}$ is mapped to an arc connecting $(u, \mathbf{r})$ to the post condition of e, i.e., $(e, *)$.

Example 18: Consider the rPES $\mathsf{P_1} = (E, U, <, \#, \prec, \triangleright)$ in Example 4. Note that $(E, <, \#)$ and its associated CN are the ones discussed in Example 15. Then, $\mathcal{A}_r(\mathsf{P_1})$, which is shown in Figure 7, is obtained by extending the CN in Figure 6 with the transitions $(b, \mathbf{r})$ and $(c, \mathbf{r})$ corresponding to the reversing events $\underline{b}$ and $\underline{c}$. The inhibitor arcs from $(b, *)$ to $(b, \mathbf{r})$ and $(c, *)$ to $(c, \mathbf{r})$ indicate that the backward transitions are enabled only after the corresponding forward transitions are fired. The inhibitor arc from $(*, c)$ to $(b, \mathbf{r})$ indicates that b cannot be fired if c has been executed.

The rCN $\mathcal{A}(\mathsf{P_3})$ associated with the rPES $\mathsf{P_3}$ in Example 8 is depicted in Figure 9. Note the absence of the place $(\{a, c\}, \#)$ (because a and c are not in conflict in $\mathsf{P_3}$) and the inhibitor arc from $(*, c)$ to $(b, \mathbf{r})$ (since b can be reversed even when $(*, c)$ is marked). Moreover, the inhibitor arc from $(d, *)$ to $(c, \mathbf{r})$ prevents c to be reversed until d is fired.

The encoding of the rPES $\mathsf{P_2}$ in Example 5 produces a net isomorphic to the one shown in Figure 7 (just places and transitions are named differently).

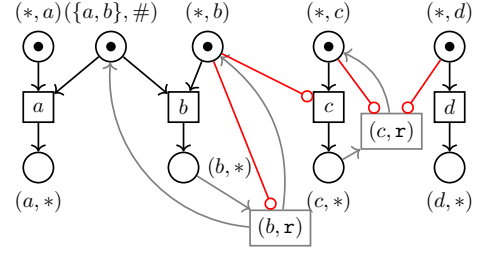*Proposition 5:* If P is an rPES then $\mathcal{A}_r(\mathsf{P})$ is an rCN.

C. From rCNes to rPESes

The encoding of rCNs into rPESes is given below.

Definition 16: Let $V^{\underline{T}} = \langle S, T, F, I, \mathsf{m} \rangle$ be an rCN with backward transitions $\underline{T}$. Then, $\mathcal{Q}_r(V^{\underline{T}}) = (E, U, <, \#, \prec, \triangleright)$ where:

1) $E = T \setminus \underline{T}$;
2) $U = \{t \mid \underline{t} \in \underline{T} \wedge t \in T \wedge {}^\bullet t = \underline{t}^\bullet\}$;
3) $< = \ll_{|E \times E}$;
4) $\# = \{(e, e') \mid e \neq e' \in E \ \wedge \ e \natural e'\}$;
5) $\prec = \{(t, \underline{t}') \mid t \in E \wedge \underline{t}' \in \underline{T} \wedge {}^\bullet t \cap {}^\circ \underline{t}' \neq \emptyset\}$;
6) $\triangleright = \{(t, \underline{t}') \mid t \in E \wedge \underline{t}' \in \underline{T} \wedge t^\bullet \cap {}^\circ \underline{t}' \neq \emptyset\}$.

The construction maps an rCN $V^{\underline{T}}$ to an rPES whose events are the forward transitions $T \setminus \underline{T}$ of $V^{\underline{T}}$. Only the forward transitions that have a reversing transition in $\underline{T}$ are undoable events (i.e, they belong to U). The causality $<$ and conflict $\#$ relations are obtained as the proper restrictions of those defined on $V^{\underline{T}}$. Finally, the reverse causation $\prec$ and the prevention $\triangleright$ are recovered from inhibitor arcs.

Example 19: Consider the rCN in Figure 7. The events of the associated rPES are $\{a, b, c, d\}$, from which $\{b, c\}$ are reversible. The only causal dependency is $b \ll c$. Conflicts are induced from shared places in the presets of transitions, which in this case are s_0 and s_2. Hence, $a \# c$ and $b \# d$. The inhibitor arc connecting s_3 to $\underline{b}$ is mapped to the reverse causality $b \prec \underline{b}$ because s_3 belongs to the preset of b. Analogously, the arc from s_4 to $\underline{c}$ (resp., from s_5 to $\underline{c}$) is mapped to $c \prec \underline{c}$ (resp., $d \prec \underline{c}$). On the contrary, the arc from s_8 to $\underline{b}$ gives $c \triangleright \underline{b}$ because s_8 belongs to the postset of c. Hence, the obtained rPES is the one defined in Example 5.

Proposition 6: If $V^{\underline{T}}$ is an rCN then $\mathcal{Q}_r(V^{\underline{T}})$ is an rPES.

D. rCNes and rPESes correspondence

We show that the mappings introduced in the previous sections establish a tight correspondence between rPESes and rCNs in terms of configurations.

Definition 17: Let $V^{\underline{T}} = \langle S, T, F, I, \mathsf{m} \rangle$ be an rCN. A *pre-configuration* X of $V^{\underline{T}}$ is a conflict-free set of forward transitions, i.e., $X \subseteq T \setminus \underline{T}$ and $\forall t, t' \in X. {}^\bullet t \cap {}^\bullet t' \neq \emptyset \Rightarrow t = t'$. Its associated marking is $m_X = \mathsf{m} - {}^\bullet X + X^\bullet$.

If X is pre-configuration of $V^{\underline{T}}$, observe that m_X may not be a reachable marking. For instance, the pre-configuration $\{c, d\}$ of the rCN in Figure 7 has the associated reachable marking $m_X = \{s_1, s_2, s_3, s_8, s_9\}$ which is not reachable because c cannot be fired if s_3 is marked.

Definition 18: A pre-configuration X of an rCN $V^{\underline{T}}$ is a *configuration* if m_X is a reachable marking of $V^{\underline{T}}$. We write $\mathsf{Conf}_{r\text{CN}}(V^{\underline{T}})$ for the set of all configurations of $V^{\underline{T}}$.

The remaining results in this section establish the operational correspondence between rPESes (resp., rCNs) and their associated rCNs (resp., rPESes). We start by showing that a set of events is enabled at a configuration X of P if and only if the corresponding set of transitions is enabled at the marking m_X of $\mathcal{A}_r(\mathsf{P})$.

Lemma 1: Let $\mathsf{P} = (E, U, <, \#, \prec, \rhd)$ be an rPES and $X \subseteq E$ a conflict-free set of events.

- If $A \cup \underline{B}$ is enabled at X then $m_X \, [A \cup \{(u, \mathtt{r}) \mid u \in B\}\rangle$ holds in $\mathcal{A}_r(\mathsf{P})$.
- If $Y \subseteq E \cup (U \times \{\mathtt{r}\})$ is a set of transitions of $\mathcal{A}_r(\mathsf{P})$ such that $m_X \, [Y\rangle$ then $(Y \cap E) \cup \{\underline{u} \mid (u, \mathtt{r}) \in Y\}$ is enabled at X.

As a consequence of the above result, we have the operational correspondence for $\mathcal{A}_r(\_)$ as stated below.

Theorem 2: Let P be an rPES. Then, $X \in \mathsf{Conf}_{r\text{PES}}(\mathsf{P})$ iff $X \in \mathsf{Conf}_{r\text{CN}}(\mathcal{A}_r(\mathsf{P}))$.

We proceed analogously for $\mathcal{Q}_r(\_)$ to show that the encoding preserves both enabling and configurations.

Lemma 2: Let $V^{\underline{T}} = \langle S, T, F, I, \mathsf{m}\rangle$ be an rCN, $X \subseteq T \setminus \underline{T}$ a pre-configuration of $V^{\underline{T}}$, $A \subseteq T \setminus \underline{T}$ a set of events, $\underline{B} = \{\underline{t} \mid t \in B\}$ a set of reversing events, with $B \subseteq T \setminus \underline{T}$.

- If $m_X \, [A \cup \underline{B}\rangle$ then $A \cup \underline{B}$ is enabled at X in $\mathcal{Q}_r(V^{\underline{T}})$.
- If $A \cup \underline{B}$ is enabled at X in $\mathcal{Q}_r(V^{\underline{T}})$ then $m_X \, [A \cup \underline{B}\rangle$.

Theorem 3: Let $V^{\underline{T}}$ be an rCN. $X \in \mathsf{Conf}_{r\text{CN}}(V^{\underline{T}})$ iff $X \in \mathsf{Conf}_{r\text{PES}}(\mathcal{Q}_r(V^{\underline{T}}))$.

From the previous results, we conclude that rCNs are a proper counterpart of rPESes.

Theorem 4: If $V^{\underline{T}}$ is an rCN then $V^{\underline{T}} \equiv \mathcal{A}_r(\mathcal{Q}_r(V^{\underline{T}}))$. If P is an rPES then $\mathsf{P} \equiv \mathcal{Q}_r(\mathcal{A}_r(\mathsf{P}))$.

VI. Causal nets and Occurrence nets

The developments in Sections IV-A and IV-B have shown that CNs are an adequate counterpart for prime event structures (Theorem 1). We now study the connections between CNs and occurrence nets, which are standardly associated with PESes. Finally, we provide further evidence suggesting that occurrence nets cannot be easily adapted to model out-of-causal order reversibility.

We start by introducing some notions about nets that will be used throughout this section. An IPT N is *single execution* if its states are sets, i.e., $\forall X \in \mathsf{St}(N). X = [\![X]\!]$. Intuitively, each transition of a single execution net can be fired at most once in a firing sequence. Single execution nets (albeit without inhibitor arcs) are also known as *1-occurrence* nets [24], [26]. The net N_1 in Figure 4a is single execution.

An IPT is said *conflict-saturated* whenever any pair of transitions that do not appear in any firing sequence of the net have a common place in their presets.

Definition 19: Let $N = \langle S, T, F, I, \mathsf{m}\rangle$ be an IPT. Two transitions $t, t' \in T$ are in *conflict* if they do not simultaneously appear in any state of N, i.e., $\{t, t'\} \not\subseteq [\![X]\!]$ for all $X \in \mathsf{St}(N)$.

N is *conflict-saturated* whenever for all $t, t' \in T$, if t and t' are in conflict then ${}^\bullet t \cap {}^\bullet t' \neq \emptyset$.

A conflict-saturated net is one in which conflicts are characterised structurally.

Example 20: The net in Figure 10 is conflict-saturated because the pairs a and c, as well as b and c, do no appear in any configuration of the net but each pair of transitions shares a place in their preset (s_7 for a and c, and s_3 for b and c).
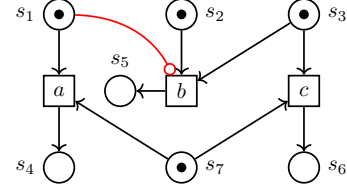

Fig. 10: A conflict-saturated net

Every single execution IPT can be transformed into a conflict-saturated, equivalent one, i.e., without changing the executions of the net. This can be achieved by adding a new place for any pair of conflicting transitions, and then extends the presets of the transitions with the corresponding new places. This procedure can be formalised as follows.

Consider a single execution IPT $N = \langle S, T, F, I, \mathsf{m}\rangle$. Define the set $\mathsf{confl}(N)$ of pairs of transitions in conflict as

$$\mathsf{confl}(N) = \{T' \subset T \mid |T'| = 2 \ \wedge \ \forall X \in \mathsf{St}(N). \ T' \not\subseteq X\}.$$

Then, the conflict-saturated version of N is given by $N^{\mathsf{confl}(N)} = \langle S', T, F', I, \mathsf{m}'\rangle$ where

- $S' = S \cup \{s_T \mid T' \in \mathsf{confl}(N)\}$,
- $F' = F \cup \{(s'_T, t) \mid t \in T'\}$;
- $\mathsf{m} = \mathsf{m} \cup \{s_T \mid T' \in \mathsf{confl}(N)\}$.

It should be noted that for each transition $t \in T$ and each reachable marking $m \in \mathcal{M}_N$ such that $m \, [t\rangle$, there exists a reachable marking $m' \in \mathcal{M}_{N^{\mathsf{confl}(N)}}$ such that $m' \, [t\rangle$ and $\forall s \in S. \ m'(s) = m(s)$, i.e., the transformation preserves the behaviour despite m and m' may differ in the added places. The correspondence holds also in the opposite direction: for each transition $t \in T$ and each reachable marking $m' \in \mathcal{M}_{N^{\mathsf{confl}(N)}}$ such that $m' \, [t\rangle$, there exists a reachable marking $m' \in \mathcal{M}_N$ such that $m' \, [t\rangle$ and $\forall s \in S. \ m'(s) = m(s)$.

The construction above gives a procedure for obtaining an equivalent, conflict-saturated version of an IPT.

Proposition 7: Let N be a single execution IPT, then there exists a conflict-saturated IPT $N^{\mathsf{confl}(N)}$ such that $N^{\mathsf{confl}(N)} \equiv N$.

Remark 1: It is worth stressing that pCNs and CNs are single execution nets; moreover, CNs are also conflict-saturated.

A. Occurrence nets

We recall the notion of *occurrence* nets[1]. We adopt the usual convention by which places and transitions are called *conditions* and *events*; and write B and E for their respective sets (instead of S and T) and c for the initial marking. We may confuse conditions with places and events with transitions. Moreover, we will omit the inhibiting relation I since occurrence nets do not have inhibitor arcs (i.e., $I = \emptyset$).

Definition 20: An *occurrence net* (ON) $O = \langle B, E, F, c \rangle$ is an acyclic, safe net satisfying the following restrictions:

- $\forall b \in B$. ${}^{\bullet}b$ is either empty or a singleton, and $\forall b \in c$. ${}^{\bullet}b = \emptyset$,
- $\forall b \in B$. $\exists b' \in c$ such that $b' \leq_O b$,
- for all $e \in E$ the set $\{e' \in E \mid e' \leq_O e\}$ is finite, and
- $\#$ is an irreflexive and symmetric relation defined as follows:
  - $e \#_0 e'$ iff $e, e' \in E$, $e \neq e'$ and ${}^{\bullet}e \cap {}^{\bullet}e' \neq \emptyset$,
  - $x \# x'$ iff $\exists y, y' \in E$ such that $y \#_0 y'$ and $y \leq_O x$ and $y' \leq_O x'$.

Each condition b in an ON represents the occurrence of a token. Hence, b either belongs to the initial marking c or is produced by the *unique* event in ${}^{\bullet}b$. The flow relation is interpreted as the *causality* relation among the elements of the net; for this reason we say that x *causally depends* on y iff $y \leq_O x$. Observe that $\leq_O \cap (E \times E)$ is a partial order and $\#$ is inherited along $\leq_O$; hence, $e \# e' \leq_O e''$ implies $e \# e''$.

Proposition 8: Let $O = \langle B, E, F, c \rangle$ be an occurrence net. Then, O is a single execution net.

As for Proposition 7, every single execution IPT can be converted into an equivalent conflict-saturated net. Since the construction does not modify the inhibitor relation, the same holds for occurrence nets.

Definition 21: Let $O = \langle B, E, F, c \rangle$ be an ON. A set of events $X \subseteq E$ is a *configuration* of O if:

- $\forall e, e' \in X. e \neq e' \Rightarrow \neg(e \# e')$, i.e., it is *conflict-free*; and
- $\forall e \in X. \lfloor e \rfloor \subseteq X$, i.e., it is *left-closed*.

The set of configurations of O is denoted by $\mathsf{Conf}_{\mathrm{ON}}(O)$. If a ON is conflict-saturated then the first condition can be rewritten as $\forall e, e' \in X. {}^{\bullet}e \cap {}^{\bullet}e' \neq \emptyset \Rightarrow e = e'$. Moreover, observe that each configuration of an ON is also a state.

B. From ON to CN

We show that every occurrence net can be turned into a causal one, and hence that the latter notion is a conservative extension of the former one.

Definition 22: Given an occurrence net $O = \langle B, E, F, c \rangle$ we can associate to it a net $\mathcal{O}(O)$ defined as $\langle S, E, F', I, m \rangle$ where

[1]Occurrence nets are often the result of the *unfolding* of a (safe) net. In this perspective an occurrence net is meant to describe precisely the non-sequential semantics of a net, and each reachable marking of the occurrence net corresponds to a reachable marking of the unfolded net. Here we focus purely on occurrence nets and not on the nets they are the unfoldings of, nor on the relation between the net and its unfolding.

- $S = \{(*, e) \mid e \in E\} \cup \{(e, *) \mid e \in E\}$
$$\cup \{(\{e, e'\}, \#) \mid e \# e'\};$$
- $F' = \{(s, e) \mid s = (*, e) \vee (s = (W, \#) \wedge e \in W)\};$
$$\cup \{(e, s) \mid s = (e, *)\};$$
- $I = \{(s, e) \mid s = (*, e') \wedge e' <_C e\};$ and
- $m : S \to \mathbb{N}$ is such that $m(s) = 0$ if $s = (e, *)$ and $m(s) = 1$ otherwise,

The construction resembles the one from PESes to CNs: each event e of an occurrence net is associated with an homonymous transition and two places $(*, e)$ and $(e, *)$ in the corresponding causal net. When marked, $(*, e)$ represents the fact that e has not been fired yet, while $(e, *)$ describes the fact that e has been executed. Moreover, there is one additional place $(\{e, e'\}, \#)$ for any pair of conflicting events $e \# e'$ in O. As expected, the preset of an event e consists of the place $(*, e)$ and all the places of the form $(\{e, e'\}, \#)$, which represent the conflicts of e with some other event e'. Causal dependencies are mapped into inhibitor arcs so that $(e', *)$ belongs to the inhibitor set of e only if e causally depends on e' in O. The initial marking assigns a token to every place belonging to the the preset of some transition.

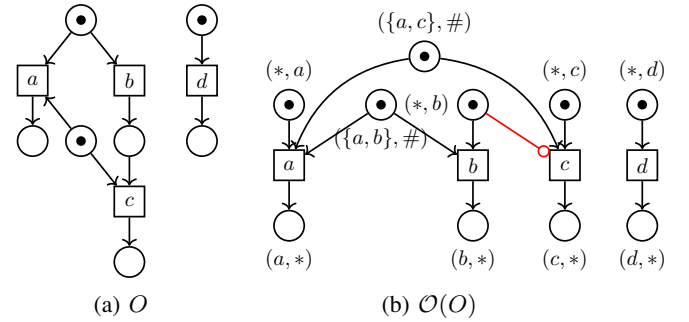*Example 21:* Figure 11 shows the encoding of the ON O in Figure 11a as the CN $\mathcal{O}(O)$ in Figure 11b.

Fig. 11: An occurrence net as a causal net

Proposition 9: Let O be an occurrence net. Then $\mathcal{O}(O)$ is a CN and $O \equiv \mathcal{O}(O)$.

We stress that $\mathcal{O}(O)$ is conflict-saturated for any ON O.

C. From pCN to ON

The encoding from pCNs to ONs basically requires to embed causal dependencies expressed in terms of inhibitor arcs into the the flow relation. For this reason, the encoding incorporates new places of the form (t, t') as a way for representing the dependency $t \lessdot t'$.

Definition 23: Let $C = \langle S, T, F, I, m \rangle$ be a pCN. The associated ON is a net $\mathcal{Z}(C) = \langle B, T, F', c \rangle$ where

- $B = S \cup \{(t, t') \mid t \lessdot t'\};$
- $F' = F \cup \{(s, t) \mid s = (t', t)\} \cup \{(t, s) \mid s = (t, t')\};$ and
- $c : B \to \mathbb{N}$ is such that $c(b) = m(b)$ if $b \in S$ and $c(b) = 0$ if $b \notin S$.

Note that the flow relation is extended to account for causal dependencies: each transition t produces tokens in the places (t, t') (i.e., $(t, (t, t')) \in F$) and consumes tokens from (t'', t)

(i.e., $(t, (t'', t))$). In this way, a transition t is enabled when all its causes have been already fired (i.e., the places (t'', t) are marked).

The following results highlights that the transformation preserves the behaviour of the net.

Proposition 10: Let C be a pCN, then $\mathcal{Z}(C)$ is an ON and $C \equiv \mathcal{Z}(C)$.

The net $\mathcal{Z}(C)$ in not necessarily conflict-saturated, as C could be not conflict-saturated.

Example 22: In Figure 12 we depict a pCN C (on the left) and the associated ON (on the right).
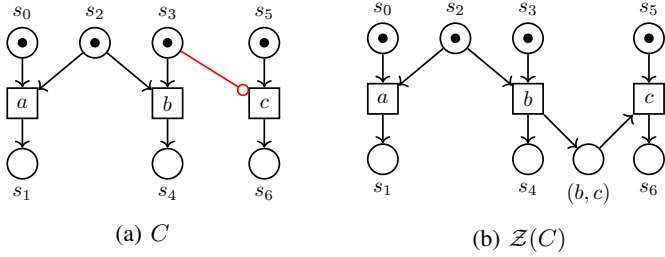

(a) C

(b) $\mathcal{Z}(C)$

Fig. 12: A causal net as an occurrence net

By combining Proposition 9 and Proposition 10 we obtain the following result.

Theorem 5: Let C be a causal net. Then $C = \mathcal{O}(\mathcal{Z}(C))$. Let O be an occurrence net. Then $O \equiv \mathcal{Z}(\mathcal{O}(O))$.

Despite we can directly obtain a PES from an ON (and vice versa) [27], the mappings introduced in Sections IV-A and IV-B give us an alternative procedure that relies on the intermediate representation of PESes in terms of CNs, as stated below.

Theorem 6: Let P be a PES. Then $\mathcal{Z}(\mathcal{A}(\mathsf{P}))$ is an ON and $\mathsf{Conf}_{\mathrm{PES}}(\mathsf{P}) = \mathsf{Conf}_{\mathrm{ON}}(\mathcal{Z}(\mathcal{A}(\mathsf{P})))$. Let O be an ON. Then $\mathcal{Q}(\mathcal{O}(O))$ is a PES and $\mathsf{Conf}_{\mathrm{ON}}(O) = \mathsf{Conf}_{\mathrm{PES}}(\mathcal{Q}(\mathcal{O}(O)))$.

D. Reversible occurrence net

A causal-consistent reversible version of ONs has be proposed in [14], [15]. The model associates each reversible event of an occurrence net with a *reversing* transition, analogously to our rCNs. As already noticed in [15], this approach fails when considering the full generality of rPES due to the tight correspondence between the configurations of an occurrence net and their associated reachable markings: given an ON $O = \langle B, E, F, \mathsf{c} \rangle$ and a configuration X, its associated reachable marking m_X can be computed as $(\mathsf{c} \cup \llbracket X^\bullet \rrbracket) \setminus \llbracket {}^\bullet X \rrbracket$ This equation makes clear that some *effects* (i.e., the produced tokens) of the executed events may not be *observable* on the reachable marking, i.e., if $e \in X$ and $e^\bullet \cap {}^\bullet X \neq \emptyset$, then $e^\bullet \nsubseteq m_X$. This is the case for the configuration $\{b, c\}$ of the ON depicted in Figure 12b, where the condition (b, c) is *used* by the firing of c, and does not hold after c is executed. Assume that we expect b to be an out-of-causal-order reversible event and add the reversing transition $\underline{b}$ that consumes from s_4 and (b, c). Then it is clear that $\underline{b}$ is not enabled at m_X; hence, the undoing of b would require the undoing of c, which imposes a causally-order reversibility model.
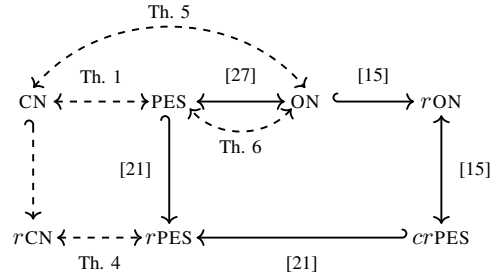

Fig. 13: Recap of results. Solid arrows represent known results/works in the literature while dashed arrows stand for results in this paper. Double-headed arrows are used for correspondences while curly tailed arrows are for inclusions and reversible extensions.

VII. Conclusions

The main contribution of this paper is the characterisation of a class of Petri nets with inhibitor arcs, dubbed *reversible causal nets* (rCN), which provides an operationally counterpart for the reversibility model behind reversible prime event structures (rPES) [21]. The key observation is that three out of the four (primitive) relations that rPESes define over events can be captured by inhibitor arcs: both *forward and reverse causality* can be modelled as inhibitor arcs that connect an event (either forward or reversing) to (some of the conditions in) the presets of its immediate causes, while *prevention* is translated to arcs that connect an event to (some of the conditions in) the postsets of the preventing events. The remaining *conflict relation* corresponds, as usual, to overlapped presets.

Figure 13 briefly recaps our results with respect to the state of the art. The correspondence between occurrence nets and prime event structures is due to Winskel's seminal work [27]. We have introduced a new class of Petri nets, called *causal nets*, and showed their correspondence with prime event structures (Theorem 1). We have then presented the reversible variant of causal nets (rCNs), and showed that they are an operational counterpart of the reversible prime event structures of [21] (Theorem 4). Finally, we showed that causal nets are tightly connected to occurrence nets (Theorem 5), and give a revisitation of Winskel's correspondence via Theorem 6. To give a complete picture of known results, we depict the extension of occurrence nets into reversible occurrence nets proposed in [15]; where the latter ones correspond to a proper subclass of rPES that just accounts for causally-consistent reversibility (crPES).

Besides contributing to the well-established line of work initiated in [18], [27] that links classes of nets with classes of event structures, our work explains the reversing mechanisms of rPESes by relying on a well-studied concurrent model. In this respect, our contribution constitutes a step forward with respect to [15], which just deals with the subclass of *causal-reversible* rPESes which, e.g., do not provide out-of-causal-order reversibility. Remarkably, our reversible causal nets do not appeal to original elements to achieve reversibility as, e.g.,

the *bonds* of [20]. As a matter of fact, causal nets do not rely either on the information (i.e., colours) carried on by tokens (as done in [3], [14]).

The characterisation of rPESes as restrictions over the structure of the nets enables the study of less constrained models. For instance, an undoable event in an rPES can be undone just by performing one reversing event. This excludes the possibility of defining alternative conditions for reversing an event. We can overcome this limitation in the operational model just by dropping the requirement of non-overlapping backward transitions (i.e., $[\![^{\bullet}\underline{T}]\!] = {}^{\bullet}\underline{T}$). This kind of analysis seems an interesting line to explore in future work.

Causal nets and their reversible versions play a role analogous to occurrence nets (in their connection with PESes). Consequently, we may ask ourselves about the class of (reversible) nets whose unfoldings are (reversible) causal nets, or, alternatively, which are the suitable categories that would allow us, if possible, to extend Winskel's chain of coreflection to reversible event structures.

REFERENCES

[1] B. Aman, G. Ciobanu, R. Glück, R. Kaarsgaard, J. Kari, M. Kutrib, I. Lanese, C. A. Mezzina, L. Mikulski, R. Nagarajan, I. C. C. Phillips, G. M. Pinna, L. Prigioniero, I. Ulidowski, and G. Vidal. Foundations of reversible computation. In I. Ulidowski, I. Lanese, U. P. Schultz, and C. Ferreira, editors, *Reversible Computation: Extending Horizons of Computing - Selected Results of the COST Action IC1405*, volume 12070 of *Lecture Notes in Computer Science*, pages 1–40. Springer, 2020.

[2] P. Baldan, N. Busi, A. Corradini, and G. M. Pinna. Domain and event structure semantics for Petri nets with read and inhibitor arcs. *Theoretical Computer Science*, 323(1-3):129–189, 2004.

[3] K. Barylska, A. Gogolinska, L. Mikulski, A. Philippou, M. Piatkowski, and K. Psara. Reversing computations modelled by coloured petri nets. In W. M. P. van der Aalst, R. Bergenthum, and J. Carmona, editors, *Proceedings of the International Workshop on Algorithms & Theories for the Analysis of Event Data 2018 Satellite event of the conferences: 39th International Conference on Application and Theory of Petri Nets and Concurrency Petri Nets 2018 and 18th International Conference on Application of Concurrency to System Design ACSD*, volume 2115 of *CEUR Workshop Proceedings*, pages 91–111. CEUR-WS.org, 2018.

[4] G. Boudol. Flow Event Structures and Flow Nets. In I. Guessarian, editor, *Semantics of Systems of Concurrent Processes*, volume 469 of *Lecture Notes in Computer Science*, pages 62–95. Springer, 1990.

[5] G. Boudol. Flow event structures and flow nets. In I. Guessarian, editor, *Semantics of Systems of Concurrent Processes*, volume 469 of *Lecture Notes in Computer Science*, pages 62–95. Springer, 1990.

[6] G. Casu and G. M. Pinna. Flow unfolding of multi-clock nets. In G. Ciardo and E. Kindler, editors, *Petri Nets 2014*, volume 8489 of *Lecture Notes in Computer Science*, pages 170–189. Springer, 2014.

[7] G. Casu and G. M. Pinna. Petri nets and dynamic causality for service-oriented computations. In A. Seffah, B. Penzenstadler, C. Alves, and X. Peng, editors, *Proceedings of SAC 2017*, pages 1326–1333. ACM, 2017.

[8] V. Danos and J. Krivine. Transactions in RCCS. In M. Abadi and L. de Alfaro, editors, *Concurrency Theory, 16th International Conference, CONCUR 2005*, volume 3653 of *Lecture Notes in Computer Science*, pages 398–412. Springer, 2005.

[9] S. Kuhn, B. Aman, G. Ciobanu, A. Philippou, K. Psara, and I. Ulidowski. Reversibility in chemical reactions. In I. Ulidowski, I. Lanese, U. P. Schultz, and C. Ferreira, editors, *Reversible Computation: Extending Horizons of Computing - Selected Results of the COST Action IC1405*, volume 12070 of *Lecture Notes in Computer Science*, pages 151–176. Springer, 2020.

[10] I. Lanese, M. Lienhardt, C. A. Mezzina, A. Schmitt, and J. Stefani. Concurrent flexible reversibility. In M. Felleisen and P. Gardner, editors, *Programming Languages and Systems - 22nd European Symposium on Programming, ESOP 2013*, volume 7792 of *Lecture Notes in Computer Science*, pages 370–390. Springer, 2013.

[11] I. Lanese, A. Palacios, and G. Vidal. Causal-consistent replay debugging for message passing programs. In J. A. Pérez and N. Yoshida, editors, *Formal Techniques for Distributed Objects, Components, and Systems - 39th IFIP WG 6.1 International Conference, FORTE 2019*, volume 11535 of *Lecture Notes in Computer Science*, pages 167–184. Springer, 2019.

[12] R. Langerak. Bundle event structures: A non-interleaving semantics for lotos. In M. Diaz and R. Groz, editors, *FORTE '92 Conference Proceedings*, volume C-10 of *IFIP Transactions*, pages 331–346. North-Holland, 1993.

[13] D. Medic, C. A. Mezzina, I. Phillips, and N. Yoshida. Towards a formal account for software transactional memory. In I. Lanese and M. Rawski, editors, *Reversible Computation - 12th International Conference, RC 2020*, volume 12227 of *Lecture Notes in Computer Science*, pages 255–263. Springer, 2020.

[14] H. Melgratti, C. A. Mezzina, and I. Ulidowski. Reversing Place Transition Nets. *Logical Methods in Computer Science*, Volume 16, Issue 4, Oct. 2020.

[15] H. C. Melgratti, C. A. Mezzina, I. Phillips, G. M. Pinna, and I. Ulidowski. Reversible occurrence nets and causal reversible prime event structures. In I. Lanese and M. Rawski, editors, *Reversible Computation - 12th International Conference, RC 2020*, volume 12227 of *Lecture Notes in Computer Science*, pages 35–53. Springer, 2020.

[16] C. A. Mezzina, R. Schlatte, R. Glück, T. Haulund, J. Hoey, M. H. Cservenka, I. Lanese, T. Æ. Mogensen, H. Siljak, U. P. Schultz, and I. Ulidowski. Software and reversible systems: A survey of recent activities. In I. Ulidowski, I. Lanese, U. P. Schultz, and C. Ferreira, editors, *Reversible Computation: Extending Horizons of Computing - Selected Results of the COST Action IC1405*, volume 12070 of *Lecture Notes in Computer Science*, pages 41–59. Springer, 2020.

[17] U. Montanari and F. Rossi. Contextual nets. *Acta Informatica*, 32(6), 1995.

[18] M. Nielsen, G. Plotkin, and G. Winskel. Petri Nets, Event Structures and Domains, Part 1. *Theoretical Computer Science*, 13:85–108, 1981.

[19] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information (10th Anniversary edition)*. Cambridge University Press, 2016.

[20] A. Philippou and K. Psara. Reversible computation in petri nets. In J. Kari and I. Ulidowski, editors, *Reversible Computation - 10th International Conference, RC 2018*, volume 11106 of *Lecture Notes in Computer Science*, pages 84–101. Springer, 2018.

[21] I. Phillips and I. Ulidowski. Reversibility and asymmetric conflict in event structures. *Journal of Logic and Algebraic Methods in Programming*, 84(6):781–805, 2015.

[22] I. Phillips, I. Ulidowski, and S. Yuen. A reversible process calculus and the modelling of the ERK signalling pathway. In R. Glück and T. Yokoyama, editors, *Reversible Computation, 4th International Workshop, RC 2012. Revised Papers*, volume 7581 of *Lecture Notes in Computer Science*, pages 218–232. Springer, 2013.

[23] G. M. Pinna. How much is worth to remember? a taxonomy based on Petri Nets Unfoldings. In L. M. Kristensen and L. Petrucci, editors, *PETRI NETS 2011*, volume 6709 of *Lecture Notes in Computer Science*, pages 109–128, 2011.

[24] R. J. van Glabbeek and G. D. Plotkin. Configuration structures. In *LICS '95*, pages 199–209. IEEE Computer Society Press, June 1995.

[25] R. J. van Glabbeek and G. D. Plotkin. Event structures for resolvable conflict. In J. Fiala, V. Koubek, and J. Kratochvíl, editors, *MFCS'04 Conference Proceedings*, volume 3153 of *Lecture Notes in Computer Science*, pages 550–561. Springer, 2004.

[26] R. J. van Glabbeek and G. D. Plotkin. Configuration structures, event structures and Petri nets. *Theoretical Computer Science*, 410(41):4111–4159, 2009.

[27] G. Winskel. Event Structures. In W. Brauer, W. Reisig, and G. Rozenberg, editors, *Petri Nets: Applications and Relationships to Other Models of Concurrency*, volume 255 of *Lecture Notes in Computer Science*, pages 325–392. Springer, 1986.