

Building Tight Occurrence Nets from Reveals Relations.

Sandie Balaguer, Thomas Chatain, Stefan Haar

▶ **To cite this version:**

Sandie Balaguer, Thomas Chatain, Stefan Haar. Building Tight Occurrence Nets from Reveals Relations.. ACSD 2011, Jun 2011, Newcastle, United Kingdom. pp.44-53, 10.1109/ACSD.2011.16 . inria-00638232

HAL Id: inria-00638232
https://inria.hal.science/inria-00638232

Submitted on 4 Nov 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Building Tight Occurrence Nets from Reveals Relations

Sandie Balaguer, Thomas Chatain, Stefan Haar
INRIA & LSV (CNRS & ENS Cachan)
61, avenue du Prsident Wilson
94235 CACHAN Cedex, France
{*balaguer, chatain, haar*}*@lsv.ens-cachan.fr*

***Abstract*—Occurrence nets are a well known partial order model for the concurrent behavior of Petri nets. The causality and conflict relations between events, which are explicitly represented in occurrence nets, induce logical dependencies between event occurrences: the occurrence of an event e in a run implies that all its causal predecessors also occur, and that no event in conflict with e occurs. But these structural relations do not express all the logical dependencies between event occurrences in maximal runs: in particular, the occurrence of e in any maximal run may imply the occurrence of another event that is not a causal predecessor of e, in that run. The *reveals* relation has been introduced in [1] to express this dependency between two events. Here we generalize the reveals relation to express more general dependencies, involving more than two events, and we introduce ERL logic to express them as boolean formulas. Finally we answer the synthesis problem that arises: given an ERL formula φ, is there an occurrence net $\mathcal{N}$ such that φ describes exactly the dependencies between the events of $\mathcal{N}$?**

***Keywords*-synthesis of concurrent systems, occurrence nets, event logics, Petri nets, maximal runs**

I. Introduction

Partial order representations of runs of Petri nets provide an alternative to sequential semantics, exhibiting the concurrency that naturally arises from the Petri net dynamics. *Occurrence nets* are the data structure for the partial order semantics referred to as *unfoldings*; they are nets in which all transitions, called *events*, are executable and the flow relation induced by the arcs is acyclic. Paths between events represent *causality*.

The representation of all runs of a Petri net as an *unfolding* [2], [3] allows one to avoid the state-space explosion due to interleavings when exploring the runs of a Petri net. Unfoldings are infinite in general, but can be represented efficiently by a finite complete prefix [4], [5], for instance to check LTL formulas [6].

The structure of an occurrence net induces three relations over its events, *causality*, *concurrency* and *conflict*, thus generating a *prime event structure* [2]. Causality represents the partial ordering of events due to the progress of the run. When two events may occur in the same run, but are not related by causality, they are concurrent. The last possibility is that two events never occur in the same run; then they are in conflict. The causality and conflict relations induce logical dependencies between event occurrences: the occurrence of an event e in a run implies that all its causal predecessors also occur, and that no event in conflict with e ever occurs.

Here, we focus on a particular setting where weak fairness [7] is assumed, i.e. any enabled event has to occur or to be disabled, and when we consider these *maximal* runs, the structural relations do not express all the logical dependencies between event occurrences. Indeed, in this context, concurrency does not necessarily mean logical independency: it is possible that the occurrence of an event implies the eventual occurrence of another one, which is structurally concurrent. This happens with events a and c in Fig. 2(a): we have to observe that a is in conflict with b and that any maximal run contains either b or c. Therefore, if a occurs in a maximal run, then b does not occur and eventually c necessarily occurs. Yet c and a are not causally related.

Another case is illustrated by events a and d in the same figure: since a is a causal predecessor of d, the occurrence of d implies the occurrence of a; but in any *maximal* run, the occurrence of a also implies the occurrence of d because d is the only possible continuation to a and nothing can prevent it. Thus a and d are actually made *logically equivalent* by the maximal progress assumption.

The *reveals* relation between events was introduced in [1] to express these implicit dependencies between two events. Knowledge of *reveals* facilitates in particular the analysis of partially observable systems, in the context of diagnosis, testing, or verification: an event b revealed by a needs not be observable if a is, the occurrence of b can be *inferred*. The equivalence classes of events that mutually reveal each other are called *facets*; contracting facets into single events creates a *reduced* occurrence net whose set of maximal executions is in bijection with that of the initial occurrence net.

While the focus in [1] was on the binary *reveals* relation, we embed in this paper the relation in a more general logical framework. Starting from the observation that the *reveals* relation corresponds to logical implication between the occurrence of events, we consider general boolean formulas where the atoms express the occurrence of events, and introduce the *ERL* logic for capturing dependencies in occurrence nets. We then show first how to build a logical formula that describes all logical dependencies between the occurrence of events. Then we ask what are the formulas that are satisfied by all the runs of an occurrence net. An important result is that the logical dependencies between events, with the maximal

progress assumption, are not only binary: there are logical dependencies that cannot be deduced from binary dependencies. This leads us to define an extended reveals relation.

Lastly, we solve the synthesis problem that arises: given an *ERL* formula over events (or facets), does this formula describe the set of possible runs of an occurrence net? We propose a method for synthesizing an occurrence net from an *ERL* formula. As a corollary, this allows us to identify a canonical occurrence net to represent the equivalence class of all occurrence nets that have the same logical dependencies between events.

The paper is organized as follows. Section II recalls the basic definitions about Petri nets, processes and occurrence nets. Section III presents the binary reveals relation and the facets abstraction from [1]. It establishes a new result about the converse well-foundedness of the reveals relation over facets. Section IV introduces the ERL logic, capable of capturing general logical dependencies between events. ERL formulas can be interpreted with respect to a set of acceptable runs of an occurrence net; an important case is that of *maximal* runs, which gives rich dependencies, and which the last sections of the paper will focus on. Section V explains how to build an ERL formula that describes the dependencies between the events of a given occurrence net. Finally, Section VI solves the problem of synthesis of occurrence nets from ERL formulas.

II. Occurrence Nets and Maximal Runs

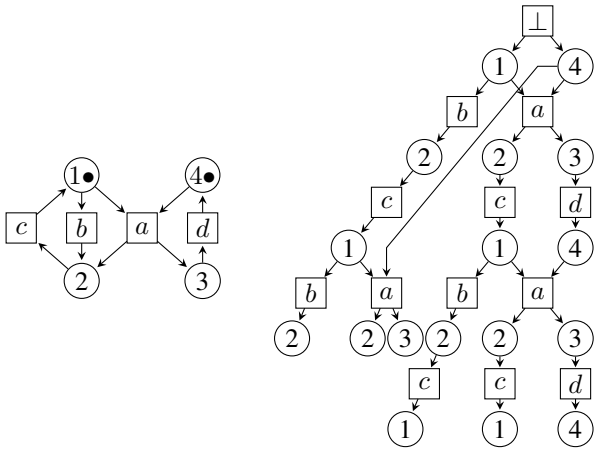
In this paper, only safe Petri nets are considered, but the results hold for bounded Petri nets also.

Definition 1 (Net)**.** A *net* is a triple (P, T, F) where P and T are disjoint sets of *places* and *transitions*, respectively, and $F \subseteq (P \times T) \cup (T \times P)$ is the *flow relation*. ▲

For any node $x \in P \cup T$, we call *pre-set* of x the set ${}^\bullet x = \{y \in P \cup T \mid (y, x) \in F\}$ and *post-set* of x the set $x^\bullet = \{y \in P \cup T \mid (x, y) \in F\}$.

A *marking* of a net is a subset of P. A *Petri net* (PN) is a tuple (P, T, F, M_0), where (P, T, F) is a finite net and $M_0 \subseteq P$ is the *initial marking*. As usual, in figures, transitions are represented as rectangles and places as circles. If $p \in M$, a black token is drawn in p. Transition t is *enabled* at M iff ${}^\bullet t \subseteq M$, i.e. t can *fire*, leading to $M' = (M \setminus {}^\bullet t) \cup t^\bullet$, in that case, we write $M \xrightarrow{t} M'$. A marking M is *reachable* if $M_0 \longrightarrow^* M$. A PN is *safe* iff for each reachable marking M, for each transition t enabled at M, $(t^\bullet \cap M) \subseteq {}^\bullet t$.

We denote by $\leq$ the *causality* relation defined as: for any transitions s and t, $s \leq t \overset{def}{\Leftrightarrow} s\ F^*\ t$, and by $<$ the corresponding strict relation. For any transition t, the set $\lceil t \rceil \overset{def}{=} \{s \mid s \leq t\}$ is the *causal past* or *prime configuration* of t, and for $T' \subseteq T$, the causal past of T' is defined as $\lceil T' \rceil \overset{def}{=} \bigcup_{t \in T'} \lceil t \rceil$. Two distinct transitions s and t are in *direct conflict*, denoted by $s \#_d t$, iff ${}^\bullet s \cap {}^\bullet t \neq \emptyset$. Two transitions s and t are in *conflict*, denoted by $s \# t$, iff $\exists s' \in \lceil s \rceil, t' \in \lceil t \rceil : s' \#_d t'$, and the *conflict set* of t is defined as $\#[t] \overset{def}{=} \{s \mid s \# t\}$. Lastly, two transitions s and t are *concurrent*, denoted by $s\ co\ t$, iff $\neg(s \# t) \wedge \neg(s \leq t) \wedge \neg(t \leq s)$.



(a) Petri net (b) Occurrence net

Figure 1. A Petri net and a prefix of its unfolding

Definition 2 (Occurrence net)**.** An *occurrence net* (ON) is a net (B, E, F) where elements of B and E are called *conditions* and *events*, respectively, and such that:

1) $\forall e \in E, \neg(e \# e)$ (no self-conflict),
2) $\forall e \in E, \neg(e < e)$ ($\leq$ is a partial order),
3) $\forall e \in E, |\lceil e \rceil| < \infty$,
4) $\forall b \in B, |{}^\bullet b| = 1$ (no backward branching),
5) $\bot \in E$ is the only $\leq$-minimal node (event $\bot$ creates the initial conditions). ▲

Fig. 1(b) gives an example of ON. An ON can also be given as a tuple $(B, E \setminus \{\bot\}, F, \mathbf{c}_0)$, where $\mathbf{c}_0 = \bot^\bullet$ is the set of minimal conditions.

A. Branching Processes and Unfoldings.

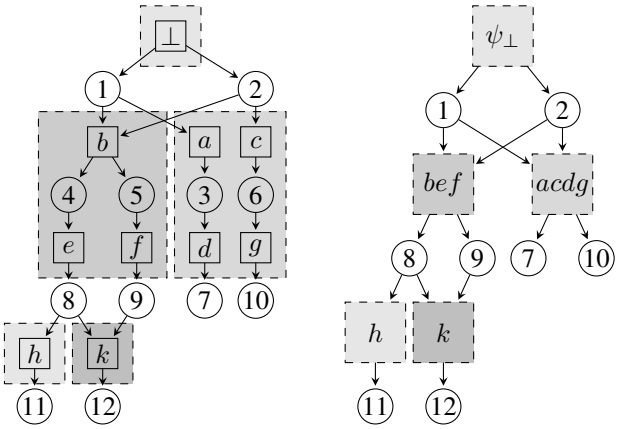
A *net homomorphism* from N to N' is a map $\pi : P \cup T \to P' \cup T'$ such that $\pi(P) \subseteq P'$, $\pi(T) \subseteq T'$, and for all $t \in T$, $\pi_{|{}^\bullet t}$, the restriction of π to ${}^\bullet t$, is a bijection between ${}^\bullet t$ and ${}^\bullet \pi(t)$, and $\pi_{|t^\bullet}$ is a bijection between $t^\bullet$ and $\pi(t)^\bullet$.

Let $N = (P, T, F, M_0)$ be a PN. A *branching process* of N is a pair (N', π), where $N' = (P', T', F', \mathbf{c}_0)$ is an ON and π is a homomorphism from (P', T', F') to (P, T, F), such that:

1) $\pi_{|\mathbf{c}_0}$ is a bijection between $\mathbf{c}_0$ and M_0,
2) $\forall t, t' \in T', \left({}^\bullet t = {}^\bullet t' \wedge \pi(t) = \pi(t')\right) \Rightarrow t = t'$

For Π_1, Π_2 two branching processes, Π_1 is a *prefix* of Π_2, written $\Pi_1 \sqsubseteq \Pi_2$, if there exists an injective homomorphism h from ON_1 into a prefix of ON_2, such that h induces a bijection between $\mathbf{c}_0^1$ and $\mathbf{c}_0^2$ and the composition $\pi_2 \circ h$ coincides with π_1.

By Theorem 23 of [2], there exists a unique (up to an isomorphism) $\sqsubseteq$-maximal branching process, called the



(a) An occurrence net. Squared events are in the same facet.

(b) The corresponding reduced ON.

Figure 2. An ON and its reduction through the facet abstraction.

unfolding of $\mathcal{N}$; by abuse of language, we will also call unfolding of $\mathcal{N}$ the ON obtained by the unfolding.

B. Properties of Maximal Runs

Definition 3 (Run, Maximal run)**.** A *run* of an ON is a conflict-free and causally closed set of events, i.e. $\omega \subseteq E$ is a run iff $\forall e \in \omega, (\#\lceil e \rceil \cap \omega = \emptyset) \wedge (\lceil e \rceil \subseteq \omega)$.
A run is *maximal* iff it is maximal w.r.t. $\subseteq$. ▲

We write Ω_{gen} for the set of all runs and Ω_{max} for the set of maximal runs.

The following lemma highlights the importance of the conflict relation in the definition of maximal runs.

Lemma 1. *A set of events ω is a maximal run iff* $\forall a \in E, a \notin \omega \Leftrightarrow \#[a] \cap \omega \neq \emptyset$.

Proof: If ω is a run and there exists $a \in E \setminus \omega$ that is not in conflict with any event of ω, then $\omega \cup \lceil a \rceil$ is also a run and ω is not maximal. Conversely, a set of events ω which satisfies the equivalence for any event a is conflict-free and $\subseteq$-maximal, and since the conflict is inherited under the causality, ω must also be causally closed. ■

III. Reveals Relation and Facets Abstraction

The structural relations $\#$, $\leq$ and co do not express all the logical dependencies between the occurrence of events in maximal runs. In particular, concurrency is not always a logical independency: it is possible that the occurrence of an event implies the occurrence of another one, which is structurally concurrent. This happens with events a and c in Fig. 2(a): we have to observe that a is in conflict with b and that any maximal run contains either b or c. Therefore, if a occurs in a maximal run, then b does not occur and eventually c necessarily occurs. Yet c and a are concurrent.

Another case is illustrated by events a and d in the same figure: because of causality, the occurrence of d implies the occurrence of a; but in any maximal run, the occurrence of a also implies the occurrence of d, because d is the only possible continuation to a and nothing can prevent it. Then a and d are actually made logically equivalent by the maximal progress assumption.

A. Reveals Relation

The reveals relation expresses dependencies between events such as "if e occurs, then f has already occurred or will occur eventually" in the sense that any run that contains e also contains f.

Definition 4 (Reveals relation [1])**.** A set of runs Ω is implicitly given. Event e *reveals* event f (in Ω), written $e \triangleright f$, iff $\forall \omega \in \Omega, (e \in \omega \Rightarrow f \in \omega)$. ▲

Notice that $\triangleright$ is transitive.

Property 1. For any events e and f, $f \leq e \Rightarrow e \triangleright f$, and if $\Omega = \Omega_{gen}$, $f \leq e \Leftrightarrow e \triangleright f$.

Proof: The implication comes directly from the fact that runs are causally closed.

For $\Omega = \Omega_{gen}$, there is no progress assumption, then for any event e, $\lceil e \rceil$ is a valid run, and consequently e does not reveal any event outside $\lceil e \rceil$. ■

Property 2 ($\#$-inheritance under $\triangleright$)**.** The conflict relation is inherited under the reveals relation: for any events a, b, c, $a \# b$ and $c \triangleright b$ together imply $a \# c$.

Proof: Assume a run contains a and c. Then, because $c \triangleright b$, it also contains b, which contradicts $a \# b$. ■

Actually, when we do not assume maximal progress, the relations between events are already given by the structural binary relations (causality and conflict). For instance, in Fig. 2(a), with the general semantics, there is no relation between the concurrent events a and c, but with the maximal semantics, for any run ω, a occurs in ω iff c occurs in ω i.e. they are not independent as in the general semantics.

In the following, we focus mainly on the maximal semantics, which gives rich dependencies between events. With this assumption, we have a nice characterization of the reveals relation based on the conflict relation. This characterization was actually used as the definition of the reveals relation in [1]. The equivalence with our definition was proved in [1].

Lemma 2 (Reveals relation: alternative definition for maximal runs)**.** *Event e* reveals *event f in Ω_{max} iff* $\#[f] \subseteq \#[e]$.

Notice that, with the general semantics, the two definitions are not equivalent. For example, in Fig. 2(a), $d \triangleright a$ holds for general runs and therefore also for maximal runs, but $a \triangleright d$ and $d \triangleright c$ hold for maximal runs only.

B. Facets Abstraction

Definition 5 (Facet [1])**.** Let $\sim$ be an equivalence relation defined as: $\forall e, f \in E, e \sim f \overset{def}{\Leftrightarrow} (e \triangleright f) \wedge (f \triangleright e)$, then a *facet* of an ON is an equivalence class of $\sim$. ▲

That is, if ψ is a facet, for any run ω and for any event e such that $e \in \psi$, $e \in \omega$ iff $\psi \subseteq \omega$. For example, in Fig. 2(a), and with the maximal semantics, the ON has five facets: $\{\bot\}$, $\{a, c, d, g\}$, $\{b, e, f\}$,$\{h\}$ and $\{k\}$.

Now we can define the causality relation, $\leq$, and the conflict relation, $\#$, over the set of facets: $\forall \psi_1, \psi_2 \in \Psi$,

$$\begin{array}{rcl} \psi_1 \leq \psi_2 & \stackrel{def}{\Leftrightarrow} & \exists e_1 \in \psi_1, e_2 \in \psi_2 : e_1 \leq e_2 \\ \psi_1 \# \psi_2 & \stackrel{def}{\Leftrightarrow} & \exists e_1 \in \psi_1, e_2 \in \psi_2 : e_1 \# e_2 \end{array}$$

We denote by $<$ the reflexive reduction of $\leq$. The set of facets equipped with $\leq$ and $\#$ is a prime event structure [1].

Reduced Occurrence Nets: For any facet and for any run, either all events in the facet are in the run or no event in the facet is in the run. Therefore, facets can be seen as events. In the sequel, we consider reduced ONs [1], i.e. ONs reduced by contracting the facets into events.

For example, in Fig. 2(a), the reduced ON is obtained by contracting, for each facet, the squared events into an event. With the maximal semantics, this gives the reduced ON of Fig. 2(b). From now on, runs are thus considered as conflict-free and causally closed sets of *facets*.

Definition 6 (Reduced occurrence net)**.** A *reduced ON* is an ON (B, Ψ, F) such that $\forall \psi_1, \psi_2 \in \Psi, \psi_1 \sim \psi_2 \Leftrightarrow \psi_1 = \psi_2$ (i.e. such that $\triangleright$ is antisymmetric). ▲

We also define the concurrency relation, co, and the reveals relation, $\triangleright$, over the set of facets: $\forall \psi_1, \psi_2 \in \Psi$,

$$\begin{array}{rcl} \psi_1 \; co \; \psi_2 & \stackrel{def}{\Leftrightarrow} & \psi_1 \neq \psi_2 \wedge \forall e_1 \in \psi_1, e_2 \in \psi_2 : e_1 \; co \; e_2 \\ \psi_1 \triangleright \psi_2 & \stackrel{def}{\Leftrightarrow} & \exists e_1 \in \psi_1, e_2 \in \psi_2 : e_1 \triangleright e_2 \end{array}$$

The results that are stated in the remaining of the article are our contribution.

Lemma 3. *In any reduced ON $\mathcal{N} = (B, \Psi, F)$ where there is no infinite set of pairwise concurrent events (in particular in the reduced unfolding of any safe Petri net), the reveals relation, $\triangleright$, is* converse well-founded *on Ψ, i.e. there is no infinite chain of distinct facets $\psi_1 \triangleright \psi_2 \triangleright \ldots$*

Proof: In the proof, we use the alternative characterization of well-foundedness: $\triangleright$ is converse well-founded on Ψ iff every nonempty subset S of Ψ has a $\triangleright$-maximal facet, i.e. a facet ψ such that for any facet $\psi' \in S$, $\psi' \neq \psi \Rightarrow \neg(\psi \triangleright \psi')$.

Assume first that the set $S \subseteq \Psi$ is conflict-free, and consider the set S' of the facets of S that have no strict causal predecessor in S. Because causality is well-founded, S' is not empty. Moreover, by definition, the facets of S' are pairwise concurrent. Thus, by hypothesis, S' is finite. Therefore there must be a facet ψ that is $\triangleright$-maximal in S'. It remains to show that ψ is also $\triangleright$-maximal in S. Let ψ' be a facet of S such that $\psi \triangleright \psi'$. By construction of S' there exists a facet ψ'' in S' such that $\psi'' \leq \psi'$. By Property 1, this implies that $\psi' \triangleright \psi''$, and by transitivity of $\triangleright$, we get $\psi \triangleright \psi''$. Since ψ is $\triangleright$-maximal in S', ψ'' must equal ψ. Then we have $\psi \triangleright \psi' \triangleright \psi$, which implies that ψ equals ψ' by construction of the facets.

If $S \subseteq \Psi$ is not conflict-free, then for any facet $\chi \in S$, the subset of S, $S_\chi = \{\chi' \in S \mid \chi \triangleright \chi'\}$ is conflict-free and hence has a $\triangleright$-maximal facet ψ. Moreover, by construction of S_χ, ψ does not reveal any facet in S, therefore, ψ is also $\triangleright$-maximal in S. ■

Anyway, Lemma 3 does *not* imply that any facet reveals only finitely many other facets. As a counterexample, consider the reduced ON of Fig. 3: facet ψ_3, associated with transition t_3, reveals all the facets $\psi_{1,i}$, $i \in \mathbb{N}^*$, associated with transition t_1.

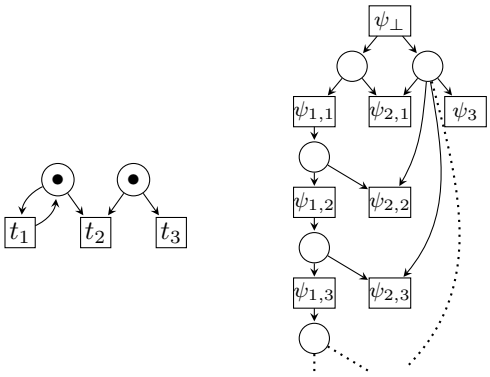


Figure 3. A Petri net and its unfolding (which is already a reduced ON)

Remark 1. For any *finite* reduced ON (B, Ψ, F), the triple $(\Psi, \triangleright^{-1}, \#)$ is a *prime event structure* [2] because:

1) $(\Psi, \triangleright^{-1})$ is a countable, partially ordered set,
2) For all $x \in \Psi$, $\{y \in \Psi \mid x \triangleright y\}$ is finite,
3) $\# \subseteq \Psi \times \Psi$ is an irreflexive and symmetric relation, and for all $x, y, z \in \Psi$, $x \# y$ and $y \triangleright z$ together imply $x \# z$ (Property 2).

C. Concurrency vs Logical Independency

Two facets may be causally ordered ($<$), in conflict ($\#$) or concurrent (co). The conflict relation exactly coincides with the fact that two facets never occur in the same execution. Moreover the causal ordering induces a reveals relation as stated in Property 1. But two concurrent facets are not necessarily logically independent in maximal runs. Hence causality and reveals together give a finer partition of the possible dependencies between two facets that are not in conflict. They can be either:

- causally related (and therefore also related by $\triangleright$),
- concurrent but related by $\triangleright$, or
- logically independent (and hence concurrent).

Formally, we define the *independency relation* among facets, denoted by ind, as follows:

$$\begin{array}{rcl} \psi_1 \; ind \; \psi_2 & \stackrel{def}{\Leftrightarrow} & \neg(\psi_1 \# \psi_2) \wedge \neg(\psi_2 \triangleright \psi_1) \wedge \neg(\psi_1 \triangleright \psi_2) \\ & \Leftrightarrow & \psi_1 \; co \; \psi_2 \wedge \neg(\psi_2 \triangleright \psi_1) \wedge \neg(\psi_1 \triangleright \psi_2) \end{array}$$

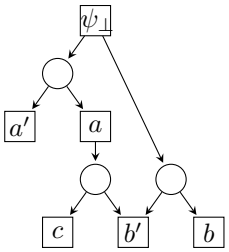


Figure 4. a *ind* b, $\neg(b$ *ind* $c)$ and $\neg(b$ *ind* $a')$

That is, two facets are independent if they are neither in conflict nor related by the reveals relation. For example, in Fig. 4, facets b and c are concurrent but not independent because c reveals b, and facets a and b are independent. Therefore, if a is in a run, this gives no information on the presence (or absence) of b in the run.

Lastly, we notice that the three relations $\triangleright$, $\#$ and *ind* are also mutually exclusive.

D. Tight (Occurrence) Nets

A tight (occurrence) net is a reduced ON in which all binary logical dependencies among facets (given by the reveals relation) are represented as causalities.

Definition 7 (Tight net)**.** A *tight net* is a reduced ON (B, Ψ, F) such that $\forall \psi_1, \psi_2 \in \Psi, \psi_1 \triangleright \psi_2 \Leftrightarrow \psi_2 \leq \psi_1$. ▲

Remark 2. In a tight net, *ind* is equivalent to *co*, and therefore the observation of the independency relation is easier than in a general reduced ON.

We will show in Section VI that it is possible to transform any finite reduced ON in a canonical tight net which accepts the same set of maximal runs Ω_{max}. This canonical tight net gives an efficient representation of the reveals relation.

IV. ERL: A Logic for Occurrence Nets

We introduce a logic, called *ERL* for *Event Reveal Logic*, that describes the properties of the runs of an ON by giving relations between event occurrences. We focus on reduced ONs and facets are used as boolean variables: ψ stands for the presence of facet ψ in a run.

We have seen that the causality relation does not explain all the dependencies betwen events of the type "if a occurs in a maximal run, then eventually b also occurs". The reveal relation was introduced to capture all these binary dependencies. But they are still not sufficient to describe more complex logical dependencies between events. Consider the reduced ON of Fig. 4: causality gives only the dependencies $a < c$ and $a < b'$, plus the trivial ones involving $\psi_\perp$. With the reveals relation we get $c \triangleright b$ and $a' \triangleright b$. They express that in any maximal run the occurrence of c implies the occurrence of b and the occurrence of a' implies the occurrence of b. But is it true that any set of facets (containing $\psi_\perp$) that satisfies these contraints, is a maximal run? The answer is no: for instance $\{\psi_\perp, a, b\}$ satisfies these constraints, but is not a valid maximal run, since c is enabled and does not occur. Actually, all the maximal runs of this ON satisfy the following constraint: if a and b occur, then c also occurs.

Our logic is designed so that it allows us to express this kind of complex dependencies between event occurrences, and to define an appropriate *extended reveals relation*.

A. Syntax and Semantics

1) Syntax: The *alphabet* consists of:

- variables: Ψ is the set of variables (including $\psi_\perp$, the facet of event $\perp$),
- constants: $\{\boldsymbol{tt}, \boldsymbol{ff}\}$
- logical connectives: $\vee, \wedge, \rightarrow, \leftrightarrow$ and $\neg$.

Well-formed formulas are called *ERL formulas* and defined inductively with the following BNF grammar:

$$\varphi ::= \boldsymbol{tt} \mid \boldsymbol{ff} \mid \psi \qquad \forall \psi \in \Psi$$
$$\mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \rightarrow \varphi \mid \varphi \leftrightarrow \varphi$$

2) Semantics: The semantics is given for a set of facets $\omega \subseteq \Psi$ and an ERL formula φ. We write $\omega \models \varphi$ when ω satisfies φ, defined as follows:

- for any facet $\psi \in \Psi$, $\omega \models \psi$ iff $\psi \in \omega$,
- the standard logical connectives $\neg$, $\vee$, $\wedge$, $\rightarrow$ and $\leftrightarrow$ have the usual semantics, in particular: $\omega \models \neg\varphi$ iff $\omega \not\models \varphi$, and $\omega \models \varphi_1 \rightarrow \varphi_2$ iff $\omega \models \varphi_1 \Rightarrow \omega \models \varphi_2$.

Since we are interested in properties of sets of runs, we look at the satisfaction of ERL formulas by sets of sets of facets: for any ERL formula φ and for any set of sets of facets Ω,

$$\Omega \models \varphi \text{ iff } \forall \omega \in \Omega, \omega \models \varphi$$

i.e. the formula is satisfied by all sets of facets. Notice that, $\Omega \not\models \varphi$ iff $\exists \omega \in \Omega : \omega \not\models \varphi$.

We define the set $[\![\varphi]\!]$ as $[\![\varphi]\!] \stackrel{def}{=} \{\omega \subseteq \Psi \mid \omega \models \varphi\}$. That is, for any $\Omega \subseteq 2^\Psi$ and for any ERL formula φ, $\Omega = [\![\varphi]\!] \Leftrightarrow (\forall \omega \in 2^\Psi, \omega \models \varphi \Leftrightarrow \omega \in \Omega)$. We write $\varphi \equiv \varphi'$ when $[\![\varphi]\!] = [\![\varphi']\!]$.

3) Extended Reveals Relation: Any well-formed formula can be brought into a *conjunctive normal form*:

$$\bigwedge_{i\in I}(b_i^1 \vee b_i^2 \vee \cdots \vee b_i^{n_i} \vee \neg a_i^1 \vee \neg a_i^1 \vee \cdots \vee \neg a_i^{m_i})$$
$$\text{iff} \quad \bigwedge_{i\in I}\big((a_i^1 \wedge a_i^2 \wedge \cdots \wedge a_i^{m_i}) \rightarrow (b_i^1 \vee b_i^2 \vee \cdots \vee b_i^{n_i})\big)$$
$$\text{iff} \quad \bigwedge_{i\in I}\Big(\bigwedge_{a\in A_i} a \rightarrow \bigvee_{b\in B_i} b\Big),$$

where $A_i = \{a_i^1, \ldots, a_i^{m_i}\}$ and $B_i = \{b_i^1, \ldots, b_i^{n_i}\}$. And since

$$\Omega \models \bigwedge_{i\in I}\Big(\bigwedge_{a\in A_i} a \rightarrow \bigvee_{b\in B_i} b\Big)$$
$$\text{iff} \quad \forall i \in I, \Omega \models \bigwedge_{a\in A_i} a \rightarrow \bigvee_{b\in B_i} b$$
$$\text{iff} \quad \forall i \in I, \forall \omega \in \Omega, A_i \subseteq \omega \Rightarrow B_i \cap \omega \neq \emptyset,$$

we can focus on formulas of the form $\bigwedge_{a\in A} a \rightarrow \bigvee_{b\in B} b$, where

A and B are two sets of facets and that are satisfied by a set of runs Ω iff whenever all facets in A occur in a run $\omega \in \Omega$, then at least one facet in B occurs in ω. This leads us to define the *extended reveals relation*.

Definition 8 (Extended reveals relation)**.** Let $\Omega \subseteq 2^{\Psi}$ be a set of runs, and A, B two sets of facets, A reveals B written $A \twoheadrightarrow B$, iff $\forall \omega \in \Omega, A \subseteq \omega \Rightarrow B \cap \omega \neq \emptyset$ ▲

In this notation, Ω becomes implicit. Notice that $\neg(A \twoheadrightarrow B)$ means $\Omega \not\models \bigwedge_{a \in A} a \rightarrow \bigvee_{b \in B} b$ i.e. $\exists \omega \in \Omega : A \subseteq \omega \wedge B \cap \omega = \emptyset$.

Remark 3. We can give a structural definition for $A \twoheadrightarrow \{b\}$: $A \twoheadrightarrow \{b\} \Leftrightarrow \#[b] \subseteq \bigcup_{a \in A} \#[a]$.

Remark 4. Conflicts can be expressed with this extended reveals relation: $\{a, b\} \twoheadrightarrow \emptyset \Leftrightarrow a \# b$.

Remark 5. The extended reveals relation is not transitive: in general $A \twoheadrightarrow B \wedge B \twoheadrightarrow C \not\Rightarrow A \twoheadrightarrow C$. Indeed, the extended reveals relation is interpreted as a conjunction of facets in the left part and as a disjunction of facets in the right part.

B. Minimal Constraints

Expressions of the form $A \twoheadrightarrow B$ are called *constraints*. We notice that some constraints can be deduced from others by monotonicity and by inheritance, which leads us to define *minimal constraints*.

1) Monotonicity Properties: First, the extended reveals relation has the following monotonicity properties:

Left Monotonicity Property. $\forall A, B, C \in 2^{\Psi}, A \twoheadrightarrow C \wedge A \subseteq B \Rightarrow B \twoheadrightarrow C$. Indeed, $A \subseteq B \Leftrightarrow \Omega \models \bigwedge_{b \in B} b \rightarrow \bigwedge_{a \in A} a$, and $\rightarrow$ is transitive.

Right Monotonicity Property. $\forall A, B, C \in 2^{\Psi}, A \twoheadrightarrow C \wedge C \subseteq B \Rightarrow A \twoheadrightarrow B$. Indeed, $C \subseteq B \Leftrightarrow \Omega \models \bigvee_{c \in C} c \rightarrow \bigvee_{b \in B} b$, and $\rightarrow$ is transitive.

Therefore, we begin by considering the constraints $A \twoheadrightarrow B$ where the sets A and B are minimal.

Definition 9 (Minimal reveals relation)**.** We define the *minimal reveals relation*, $\twoheadrightarrow_m$, as: $\forall A, B \in 2^{\Psi}$,

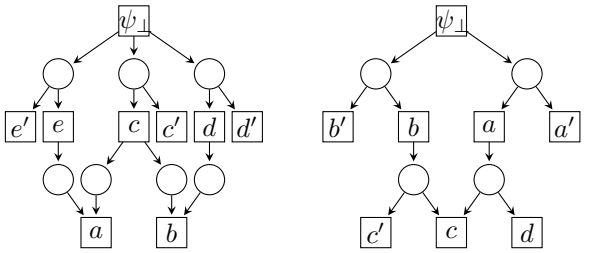
$$A \twoheadrightarrow_m B \overset{def}{\Leftrightarrow} \begin{cases} (A \neq B) \wedge (A \twoheadrightarrow B) \\ \wedge\ (\nexists B' \subsetneq B : A \twoheadrightarrow B') \\ \wedge\ (\nexists A' \subsetneq A : A' \twoheadrightarrow B) \end{cases}$$

i.e. if one facet is removed from the left part or the right part, the reveals relation is lost. ▲

For example, in Fig. 4, $\{a, b\} \twoheadrightarrow_m \{c\}$ because none of the following constraints holds: $\{a\} \twoheadrightarrow \{c\}$, $\{b\} \twoheadrightarrow \{c\}$, $\emptyset \twoheadrightarrow \{c\}$ and $\{a, b\} \twoheadrightarrow \emptyset$.

Lemma 4 (A minimal conflict is over 2 facets)**.** *For any set of facets A, $A \twoheadrightarrow_m \emptyset \Rightarrow |A| = 2$.*

Proof: Assume $A \twoheadrightarrow_m \emptyset$, then $A \neq \emptyset$ (by definition of $\twoheadrightarrow$). Moreover, for any $a \in A$, $A \setminus \{a\} \not\twoheadrightarrow \emptyset$. The set $\omega = \lceil A \setminus \{a\} \rceil$ is causally closed and conflict-free because $A \setminus \{a\} \not\twoheadrightarrow \emptyset$, therefore ω is a general run. Now consider $\chi = \omega \cup \lceil a \rceil$, $A \subseteq \chi$ and since $A \twoheadrightarrow \emptyset$, χ is not a general run even though it is causally closed, so it necessarily contains two facets b and c that are in conflict and such that $b \in \omega$ and $c \in \lceil a \rceil$. Therefore, there exists one facet $d \in A \setminus \{a\}$ (d is a successor of b) which is in conflict with a, i.e. $\{a, d\} \twoheadrightarrow \emptyset$. For any facet e, $\{e\} \twoheadrightarrow \emptyset$ means that no run contains e and contradicts the fact that all the events (here facets) of an ON are executable. Therefore, $\{a, d\} \twoheadrightarrow_m \emptyset$, and since $\{a, d\} \subseteq A$ and $A \twoheadrightarrow_m \emptyset$, we must have $A = \{a, d\}$. ■



(a) $\{c, d\} \twoheadrightarrow \{b\}$ is a minimal constraint and $\{a, d\} \twoheadrightarrow \{b\}$ is not.

(b) $\{a\} \twoheadrightarrow \{c, d\}$ is a minimal constraint and $\{a\} \twoheadrightarrow \{b, d\}$ is not.

Figure 5. Minimal constraints

2) Deduction Through a Singleton: Moreover, the following properties also hold:

Left Inheritance Property. $\forall A, B \in 2^{\Psi}$, $(A \cup \{d\} \twoheadrightarrow B) \wedge (\{d'\} \twoheadrightarrow \{d\}) \Rightarrow A \cup \{d'\} \twoheadrightarrow B$

Right Inheritance Property. $\forall A, B \in 2^{\Psi}$, $(A \twoheadrightarrow B \cup \{d\}) \wedge (\{d\} \twoheadrightarrow \{d'\}) \Rightarrow A \twoheadrightarrow B \cup \{d'\}$

Definition 10 (Immediate reveals relation)**.** We define the *immediate reveals relation*, $\twoheadrightarrow_i$, as: $\forall A, B \in 2^{\Psi}$,

$$A \twoheadrightarrow_i B \overset{def}{\Leftrightarrow} \begin{cases} A \twoheadrightarrow_m B \\ \wedge\ \forall a \in A, \nexists a' \in \Psi \setminus \{A \cup B\} : \\ \qquad (a \triangleright a' \wedge A_{a'/a} \twoheadrightarrow B) \\ \wedge\ \forall b \in B, \nexists b' \in \Psi \setminus \{A \cup B\} : \\ \qquad (b' \triangleright b \wedge A \twoheadrightarrow B_{b'/b}) \end{cases}$$

where $A_{a'/a}$ denotes $A \cup \{a'\} \setminus \{a\}$. ▲

Constraints with this immediate reveals relation are called *minimal constraints*.

For example, in Fig. 5(a), $\{a, d\} \twoheadrightarrow_m \{b\}$ is not a minimal constraint because $a \triangleright c$ and $\{c, d\} \twoheadrightarrow \{b\}$. And in Fig. 5(b) $\{a\} \twoheadrightarrow_m \{b, d\}$ is not a minimal constraint because $c \triangleright b$ and $\{a\} \twoheadrightarrow \{c, d\}$.

3) Binary Minimal Constraints: Two kinds of binary minimal constraints will be particularly useful in the sequel: those of the form $\{a, b\} \twoheadrightarrow \emptyset$ and $\{a\} \twoheadrightarrow \{b\}$.

First we define the *immediate conflict* relation, $\#_i$, as a special case of the immediate reveals relation: for all facets a and b, $\{a, b\} \twoheadrightarrow_i \emptyset \Leftrightarrow a \#_i b$. For example, in Fig. 6(c), a' and c are in conflict but not in immediate conflict because $a' \# a$ and $c \triangleright a$.

Secondly, we define the *direct reveals* relation, $\triangleright_i$, as: $a \triangleright_i b \overset{def}{\Leftrightarrow} \{a\} \rightarrowtriangle_i \{b\}$. This relation is the transitive reduction of the binary reveals relation. For example, in Fig. 6(b), $b \triangleright_i \psi_\perp$ and $\neg(c \triangleright_i \psi_\perp)$.

Remark 6. The reveals relation is the transitive and reflexive closure of the direct reveals relation. The conflict relation can be deduced by $\triangleright$-inheritance from the immediate conflict relation. Therefore, the conflict relation can be deduced from the direct reveals relation and the immediate conflict relation.

C. A Synthesis Problem for Occurrence Nets

First, in Section V, we show how to build the ERL formula $\Phi_\mathcal{N}$ which describes the set of maximal runs of a finite reduced ON $\mathcal{N}$, i.e. such that $\Omega_\mathcal{N}^{max} = [\![\Phi_\mathcal{N}]\!]$. Second, in Section VI, we present a procedure to answer whether there exists a reduced ON $\mathcal{N}$ such that its set of maximal runs is described by a given ERL formula φ.

V. From Occurrence Nets to ERL Formulas

For a given reduced ON $\mathcal{N}$, we start by building $\Phi_\mathcal{N}^{gen}$, a formula such that $[\![\Phi_\mathcal{N}^{gen}]\!] = \Omega_\mathcal{N}^{gen}$, from the characterization of general runs. Then we build $\Phi_\mathcal{N}$, a formula such that $[\![\Phi_\mathcal{N}]\!] = \Omega_\mathcal{N}^{max}$, by adding terms corresponding to the progress assumption to $\Phi_\mathcal{N}^{gen}$. The construction of $\Phi_\mathcal{N}^{gen}$ is similar to [8], where the authors build what they call "configuration constraints" also by considering the causal closure and the conflict-freeness of the configurations (or general runs).

For any facets a and b, we define the direct causality relation, $\lessdot$, as: $a \lessdot b \overset{def}{\Leftrightarrow} (a < b) \wedge (\nexists c : a < c \wedge c < b)$.

By definition, a set of facets is a general run iff it is closed under $<$ and conflict-free. That is, for a given reduced ON $\mathcal{N}$, we can build the formula $\Phi_\mathcal{N}^{gen}$ as follows:

$$\Phi_\mathcal{N}^{gen} = \underbrace{\bigwedge_{a,b\in\Psi, a<b} (b \rightarrow a)}_{\text{causal closure}} \wedge \underbrace{\bigwedge_{a,b\in\Psi, a\#b} (\neg a \vee \neg b)}_{\text{conflict-freeness}}$$

Therefore, for a given reduced ON $\mathcal{N}$, $\Phi_\mathcal{N}$ can be built as follows:

$$\begin{aligned}\Phi_\mathcal{N} \;=\; & \bigwedge_{a,b\in\Psi, a<b} (b \rightarrow a) \wedge \bigwedge_{a,b\in\Psi, a\#b} (\neg a \vee \neg b) \\ \wedge\; & \bigwedge_{a\in\Psi} \Big((\underbrace{\bigwedge_{b\in\Psi, b\lessdot a} b}_{a \text{ enabled}}) \rightarrow (a \vee \bigvee_{c\in\Psi, c\#_d a} c) \Big)\end{aligned}$$

The new part is implied by the maximality and stands for "for any facet a, if a is enabled, then a or a direct conflict with a has to fire".

Since $<$ is transitive, in the first part, we can consider only $\lessdot$, and since $\#$ is inherited through $<$, in the second part, we can consider only the direct conflict $\#_d$, and eventually:

$$\begin{aligned}\Phi_\mathcal{N} \;\equiv\; & \bigwedge_{a,b\in\Psi, a\lessdot b} (b \rightarrow a) \wedge \bigwedge_{a,b\in\Psi, a\#_d b} (\neg a \vee \neg b) \\ \wedge\; & \bigwedge_{a\in\Psi} \Big((\bigwedge_{b\in\Psi, b\lessdot a} b) \rightarrow (a \vee \bigvee_{c\in\Psi, c\#_d a} c) \Big)\end{aligned}$$

Notice that, since $\psi_\perp$ has no conflict and no causal predecessor, the third part with $a = \psi_\perp$ gives $\mathit{tt} \rightarrow \psi_\perp$ which can be reduced in $\psi_\perp$, i.e. $\psi_\perp$ is always true.

For example, in Fig. 5(b):

$$\begin{aligned}\Phi_\mathcal{N} \;\equiv\; & (c' \rightarrow b) \wedge (c \rightarrow b) \wedge (c \rightarrow a) \wedge (d \rightarrow a) \\ & \wedge (\bar{a'} \vee \bar{a}) \wedge (\bar{b'} \vee \bar{b}) \wedge (\bar{c'} \vee \bar{c}) \wedge (\bar{c} \vee \bar{d}) \\ & \wedge \psi_\perp \wedge ((a \wedge b) \rightarrow (c \vee c' \vee d)) \\ & \wedge (a \rightarrow (c \vee d)) \wedge (b \rightarrow (c' \vee c)) \\ & \wedge (\psi_\perp \rightarrow (b' \vee b)) \wedge (\psi_\perp \rightarrow (a' \vee a)),\end{aligned}$$

where $\bar{a}$ stands for $\neg a$.

We have voluntarily omitted terms of the form $a \rightarrow \psi_\perp$ that are redundant since $\psi_\perp$ must be true.

VI. From ERL formulas to Occurrence Nets: a Synthesis Procedure

The synthesis problem for PNs has been widely studied. It consists in answering whether, given a behavior, there exists a PN with this behavior. The behavior can be specified as a transition system [9]–[12] or a language, be it (i) a sequential language: in [13], the behavior is bounded by two regular languages; or (ii) a finite partial language (finite set of labeled partial orders): [14]. Most of the time, the synthesis procedure is based on the notion of region [15], [16].

In this paper, we propose another approach and we solve the following synthesis problem: given an ERL formula φ, is there a reduced ON $\mathcal{N}$ whose behavior is the one specified by φ, i.e. such that the set of maximal runs of $\mathcal{N}$, $\Omega_\mathcal{N}^{max}$, is equivalent to $[\![\varphi]\!]$?

In the sequel, we give a procedure to build a net, $\mathrm{CN}(\varphi)$, from an ERL formula φ. First, a set of binary minimal constraints is extracted from φ, then, $\mathrm{CN}(\varphi)$, is built from these constraints. If $\mathrm{CN}(\varphi)$ is a reduced ON, then $\Phi_{\mathrm{CN}(\varphi)}$ is computed and compared with φ. As in the other procedures of synthesis, places are used to restrict the behavior of the net and denote dependencies between occurrences of transitions.

A. Extracting a Set of Minimal Constraints

The set of maximal runs is given by the conflict relation which can be deduced from the direct reveals relation and the immediate conflict relation (Lemma 1 and Remark 6). Therefore, if there exists a reduced ON $\mathcal{N}$ such that $\Omega_\mathcal{N}^{max} = [\![\varphi]\!]$, then the binary minimal constraints, i.e. expressions of the form $a \triangleright_i b$ and $a \#_i b$, are enough to describe $\Omega_\mathcal{N}^{max}$ (and thus also to describe φ). That is why we focus on binary minimal constraints.

Our problem is to decide whether binary constraints of the form $a \triangleright b$ (respectively $\{a, b\} \rightarrowtriangle \emptyset$) are satisfied by φ. This amounts to decide whether $\varphi \rightarrow (a \rightarrow b)$ (respectively $\varphi \rightarrow (\neg a \vee \neg b)$) is a tautology. This problem is co-NP-

complete and can be solved quite efficiently in practice by SAT-solvers.

B. Building a Canonical Tight Net

We denote by $\Psi(\varphi)$ the set of variables that appear in φ. Each binary minimal constraint extracted from φ is represented by a condition connected to the facets that appear in the constraint. The net $\mathrm{CN}(\varphi)$ is defined as follows.

Definition 11 ($\mathrm{CN}(\varphi)$)**.** Let φ be an ERL formula. $\mathrm{CN}(\varphi) = (B, \Psi, F)$ is the finite net such that $\Psi = \Psi(\varphi)$, $B = B_1 \cup B_2$ and $F = F_1 \cup F_2$, where:

- $B_1 = \big\{\{\psi, \psi'\} \mid \psi \#_i \psi'\big\}$,
- $F_1 = \big\{(\{\psi, \psi'\}, \psi) \in B_1 \times \Psi\big\}$
  $\cup \big\{(\psi_\perp, \{\psi, \psi'\}) \in \Psi \times B_1\big\}$.

That is, for each constraint of the form $\psi \#_i \psi'$, one condition b is created and connected to $\psi_\perp$, ψ and ψ' such that ${}^\bullet b = \{\psi_\perp\}$ and $b^\bullet = \{\psi, \psi'\}$.

- $B_2 = \big\{(\psi, \psi') \in (\Psi \setminus \{\psi_\perp\})^2 \mid \psi' \rhd_i \psi\big\}$,
- $F_2 = \big\{\big((\psi, \psi'), \psi'\big) \in B_2 \times \Psi\big\}$
  $\cup \big\{\big(\psi, (\psi, \psi')\big) \in \Psi \times B_2\big\}$.

That is, for each constraint of the form $\psi' \rhd_i \psi$, one condition is created and connected to ψ and ψ' such that ${}^\bullet b = \{\psi\}$ and $b^\bullet = \{\psi'\}$. Notice that constraints of the form $\psi \rhd_i \psi_\perp$ are not considered because, if φ describes the maximal runs of a reduced ON, they are already represented by B_1 and F_1. ▲

Lemma 5. *Let $\mathcal{N}$ be a finite reduced ON, then $\mathrm{CN}(\Phi_\mathcal{N})$ is a tight net and $\Phi_{\mathrm{CN}(\Phi_\mathcal{N})} \equiv \Phi_\mathcal{N}$.*

Proof: We call $\mathcal{CN}$ the net $\mathrm{CN}(\Phi_\mathcal{N})$. We first show that $\mathcal{CN}$ is an ON, then that it is reduced, and lastly that it is a tight net. $\mathcal{N}$ and $\mathcal{CN}$ have the same conflict relation, because they have the same reveals relation and the same immediate conflict relation (Remark 6). Moreover $\mathcal{CN}$ is built so that $\forall a, b \in \Psi, a \leq_\mathcal{CN} b \Leftrightarrow b \rhd a$. Therefore, $\mathcal{CN}$ is an ON because:

- There is no self-conflict in $\mathcal{CN}$, because there is no self-conflict in $\mathcal{N}$.
- $\leq_\mathcal{CN}$ is equivalent to $\rhd^{-1}$ therefore it is a partial order.
- $\forall \psi \in \Psi, \{\psi' \mid \psi' \leq_\mathcal{CN} \psi\}$ is finite because Ψ is finite.
- There is no backward branching by construction.
- $\psi_\perp \in \Psi$ is the only minimal node by construction.

Since $\Phi_\mathcal{N}$ is associated with the reduced ON $\mathcal{N}$, it is such that, for any distinct variables $v_1, v_2 \in \Psi$, $[\![\Phi_\mathcal{N}]\!] \not\models v_1 \leftrightarrow v_2$. Therefore, $\mathcal{CN}$ is also reduced. Lastly, by construction, $\mathcal{CN}$ is a tight net.

Moreover, by Lemma 1, the set of maximal runs can be defined from the conflict relation only. $\mathcal{N}$ and $\mathcal{CN}$ have the same conflict relation. Therefore, $\mathcal{N}$ and $\mathcal{CN}$ have the same set of runs and equivalent associated ERL formulas. ■

From Lemma 5, we can derive the following theorem.

Theorem 1. *Let φ be an ERL formula, there exists a finite reduced ON $\mathcal{N}$ such that $\Phi_\mathcal{N} \equiv \varphi$ iff $\mathrm{CN}(\varphi)$ is a reduced ON and $\Phi_{\mathrm{CN}(\varphi)} \equiv \varphi$.*

Proof: ($\Rightarrow$) If there exists a reduced ON $\mathcal{N}$ such that $\Phi_\mathcal{N} \equiv \varphi$, then, by Lemma 5 $\mathrm{CN}(\varphi)$ is a candidate.

($\Leftarrow$) We choose $\mathcal{N} = \mathrm{CN}(\varphi)$. ■

Notice that $\mathcal{N}$ and $\mathrm{CN}(\Phi_\mathcal{N})$ may not accept the same general runs because some concurrent facets in $\mathcal{N}$ may be causally ordered in $\mathrm{CN}(\Phi_\mathcal{N})$. And if φ does not come from a reduced ON, the net $\mathrm{CN}(\varphi)$, obtained by the synthesis from φ, may not be a reduced ON (see Example 5). When $\mathrm{CN}(\varphi)$ is a reduced ON, it is called *canonical tight net* associated with φ (or with $\mathcal{N}$ when $\mathcal{N}$ is given).

Remark 7. In the construction, the immediate conflicts are represented by a condition connected to $\psi_\perp$. This results in a large set of initial conditions. It is possible to improve the construction by representing each immediate conflict $\psi \#_i \psi'$ by a place connected to any facet ψ_1 such that $\psi \rhd \psi_1$ and $\psi' \rhd \psi_1$. One possible choice would be to consider the $\rhd$-successors of ψ and ψ', defined as $\rhd[\psi, \psi'] = \{\psi_1 \in \Psi \mid \psi \rhd \psi_1 \wedge \psi' \rhd \psi_1\}$, create one condition b_1 for each $\rhd$-minimal facet, ψ_1, in $\rhd[\psi, \psi']$, and connect b_1 to ψ_1, ψ and ψ'. This would define B_1 and F_1. Then, any constraint of the form $\psi' \rhd_i \psi$ would be represented as previously by B_2 and F_2, except that, in B_2, we need to consider only non-redundant conditions. Indeed, if there exists $b \in B_1$ such that $(\psi, b) \in F_1 \wedge (b, \psi') \in F_1$, then $\psi' \rhd_i \psi$ is already represented and can be ignored in B_2.

C. Examples

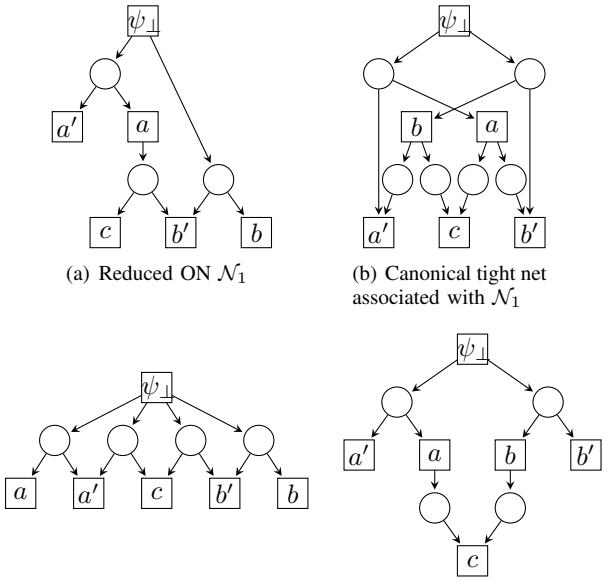
This synthesis of a tight net lets us tackle two problems.

1) Given a reduced ON $\mathcal{N}$, build the associated canonical tight net: We compute $\Phi_\mathcal{N}$ and build the net $\mathrm{CN}(\Phi_\mathcal{N})$.

Example 1. The initial reduced ON, $\mathcal{N}_1$, is depicted in Fig. 6(a). The set of maximal runs is $\Omega_{\mathcal{N}_1} = \big\{\{\psi_\perp, a, b, c\}, \{\psi_\perp, a, b'\}, \{\psi_\perp, a', b\}\big\}$ and the binary minimal constraints are $a \rhd_i \psi_\perp$, $b \rhd_i \psi_\perp$, $c \rhd_i a$, $c \rhd_i b$, $a' \rhd_i b$, $b' \rhd_i a$, $a \#_i a'$ and $b \#_i b'$. The canonical tight net obtained by the synthesis from these constraints is represented in Fig. 6(b).

Example 2. Fig. 6(c) and 6(d) give another example of a reduced ON and its associated canonical tight net. The set of maximal runs is $\Omega_{\mathcal{N}_2} = \big\{\{\psi_\perp, a, b, c\}, \{\psi_\perp, a, b'\}, \{\psi_\perp, a', b\}, \{\psi_\perp, a', b'\}\big\}$ and the binary minimal constraints are $a \rhd_i \psi_\perp$, $b \rhd_i \psi_\perp$, $c \rhd_i a$, $c \rhd_i b$, $a \#_i a'$ and $b \#_i b'$.

2) Given a formula φ, does there exist a reduced ON $\mathcal{N}$ such that $\Phi_\mathcal{N} \equiv \varphi$?: We suppose that φ is such that for any distinct variables $v_1, v_2 \in \Psi(\varphi)$, $[\![\varphi]\!] \not\models (v_1 \leftrightarrow v_2)$ (otherwise, it is possible to reduce φ by replacing each equivalence classe of variables by one variable). We extract a set of binary minimal constraints from φ and build the net $\mathrm{CN}(\varphi)$.



(a) Reduced ON $\mathcal{N}_1$

(b) Canonical tight net associated with $\mathcal{N}_1$

(c) Reduced ON $\mathcal{N}_2$

(d) Canonical tight net associated with $\mathcal{N}_2$

Figure 6. Examples of a reduced ONs with their associated canonical tight net.

Example 3. Consider the following formula:

$$\varphi = \psi_\perp \wedge (\bar{a} \vee \bar{b})$$

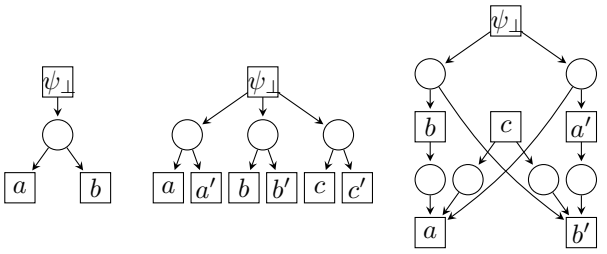
The set of runs described by φ is $[\![\varphi]\!] = \big\{\{\psi_\perp\}, \{\psi_\perp, a\}, \{\psi_\perp, b\}\big\}$. The binary minimal constraints are: $a \triangleright_i \psi_\perp$, $b \triangleright_i \psi_\perp$ and $a \#_i b$, and the ON $\mathcal{N}$ synthesized from these constraints is given in Fig. 7(a). $\mathcal{N}$ is a reduced ON but $\Omega_\mathcal{N} = \big\{\{\psi_\perp, a\}, \{\psi_\perp, b\}\big\} \neq [\![\varphi]\!]$. Therefore, there is no reduced ON $\mathcal{N}$ such that $\varphi \equiv \Phi_\mathcal{N}$. We can see that the maximality constraint $a \vee b$ is not respected by φ.

Example 4. Consider the following formula:

$$\begin{aligned}\varphi = \quad & (\psi_\perp \wedge a \wedge b \wedge \bar{c} \wedge \bar{a'} \wedge \bar{b'} \wedge c') \\ \vee\ & (\psi_\perp \wedge a \wedge \bar{b} \wedge c \wedge \bar{a'} \wedge b' \wedge \bar{c'}) \\ \vee\ & (\psi_\perp \wedge \bar{a} \wedge b \wedge c \wedge a' \wedge \bar{b'} \wedge \bar{c'})\end{aligned}$$

The set of runs described by φ is $[\![\varphi]\!] = \big\{\{\psi_\perp, a, b, c'\}, \{\psi_\perp, a, b', c\}, \{\psi_\perp, a', b, c\}\big\}$. The binary minimal constraints are: $a \#_i a'$, $b \#_i b'$, $c \#_i c'$ and for each $\psi \in \Psi \setminus \{\psi_\perp\}$, $\psi \triangleright_i \psi_\perp$. The ON $\mathcal{N}$ synthesized from these constraints is given in Fig. 7(b). $\mathcal{N}$ is a reduced ON but $\Omega_\mathcal{N} = \big\{\{\psi_\perp, a, b, c\}, \{\psi_\perp, a', b, c\}, \{\psi_\perp, a, b', c\}, \{\psi_\perp, a, b, c'\}, \{\psi_\perp, a', b', c'\}, \{\psi_\perp, a, b', c'\}, \{\psi_\perp, a', b, c'\}, \{\psi_\perp, a', b', c\}\big\} \neq [\![\varphi]\!]$. Therefore, there is no reduced ON $\mathcal{N}$ such that $\varphi \equiv \Phi_\mathcal{N}$.

Notice that this example illustrates a minimal conflict between a, b and c: $\{a, b\}$, $\{a, c\}$, and $\{b, c\}$ can occur in a run, but $\{a, b, c\}$ cannot, which is not possible in general ONs (see Lemma 4).



(a) (b) (c)

Figure 7. Examples when there is no reduced ON $\mathcal{N}$ such that $\varphi \equiv \Phi_\mathcal{N}$. (a) and (b): $\Phi_{\mathrm{CN}(\varphi)} \not\equiv \varphi$, (c): $\mathrm{CN}(\varphi)$ is not an ON.

Example 5. Consider the following formula:

$$\begin{aligned}\varphi = \quad & \psi_\perp \wedge (a \to c) \wedge (b' \to c) \wedge (b' \to a') \\ \wedge\ & (\bar{a} \vee \bar{a'}) \wedge (\bar{b} \vee \bar{b'}) \\ \wedge\ & (a \vee a') \wedge (b \vee b') \wedge (c \to (a \vee b'))\end{aligned}$$

The set of runs described by φ is $[\![\varphi]\!] = \big\{\{\psi_\perp, a, b, c\}, \{\psi_\perp, a', b', c\}, \{\psi_\perp, a', b\}\big\}$. The binary minimal constraints are: $a \triangleright_i b$, $a \triangleright_i c$, $b' \triangleright_i a'$, $b' \triangleright_i c$, $b \triangleright_i \psi_\perp$, $a' \triangleright_i \psi_\perp$, $c \triangleright_i \psi_\perp$, $a \#_i a'$ and $b \#_i b'$, and the net synthesized from these constraints is given in Fig. 7(c). We can see that this net is not an ON because there are two minimal events, c and $\psi_\perp$. Therefore, there is no reduced ON $\mathcal{N}$ s.t. $\varphi \equiv \Phi_\mathcal{N}$.

VII. Discussion

A. Results

We have shown how the structural *reveals*-relation from [1] generalizes into a framework for the description of implications between event occurrences in occurrence nets. A new logic, *ERL*, has been introduced and studied; in particular, we have solved synthesis of occurrence nets from *ERL* formulas, yielding a canonical class of reduced and *tight* nets. Even if *ERL* is a logic adapted for partial order semantics, it differs in its aim and structure from the other logics that have been proposed in the literature (for temporal logics for traces and event structures, see e.g. [17], [18]). First, *ERL* is not, strictly speaking, a *temporal* logic, since the notions of *before*, *after*, *future*, *until* etc. are of no particular relevance here; in fact, the progression of time is encapsulated in the underlying structure over which one chooses to interpret *ERL* formulas, and in the choice of admissible runs in that structure. The present paper has focused on the *maximal runs* perspective; below, we discuss extensions of the setup. Thus far, we have intended and used the *ERL* logic as a means for coding and manipulating *structure* (of occurrence nets) and *knowledge* (observing A reveals B, i.e. gives knowledge about B's occurrence). The results here open some new roads towards efficient verification of system properties, as well as towards *enforcing* such properties through behavior control, or directly through synthesis of systems from logical specifications.

B. Extensions

Moving further, several extensions are within reach. For instance, one may refine the synthesis problem above such that one specifies not only the logical relations (via an ERL formula φ), but also the causality relation that is expected.

C. General semantics

Another important line of research is to go beyond the maximal semantics considered here. As we have seen, the reveals relation depends on the set of runs that we consider. Apart from the set of maximal runs Ω_{max}, the set of all runs Ω_{gen} is another relevant choice; still more possibilities exist. With Ω_{gen}, we have already noticed that the reveals relation is given by the causality. Furthermore, we have:

Property 3 (Any minimal constraint is binary)**.** With the general semantics, for any sets of facets A and B, $A \twoheadrightarrow B \Leftrightarrow (\exists a \in A, b \in B : b \leq a) \vee (\exists a, a' \in A : a \# a')$.

Proof: s ($\Leftarrow$) If there exist $a, a' \in A$ such that $a \# a'$, then, no run contains A and for any set of facets C, $A \twoheadrightarrow C$. And if there exist $a \in A$ and $b \in B$ such that $b \leq a$, then $\{a\} \twoheadrightarrow \{b\}$ and by the monotonicity of $\twoheadrightarrow$, $A \twoheadrightarrow B$.

($\Rightarrow$) Assume $A \twoheadrightarrow B$ and A is conflict-free. Since we make no progress assumption, $\lceil A \rceil = \cup_{a \in A} \lceil a \rceil$ is a valid run. By definition of $\twoheadrightarrow$, $\forall \omega \in \Omega_{gen}$, $A \subseteq \omega \Rightarrow \omega \cap B \neq \emptyset$, and in particular, for $\omega = \lceil A \rceil$, this implies that $\lceil A \rceil \cap B \neq \emptyset$ i.e. that there exist $b \in B$ and $a \in A$ such that $b \leq a$. ■

Therefore, with general runs, non binary constraints can be decomposed as disjunctions of binary ones, in contrast to the case for Ω_{max}.

We have seen in Section V that the set of general runs can be expressed as an ERL formula. The synthesis problem can also be solved for the general semantics. More surprisingly, the procedure for solving it is exactly the same as in Section VI and Theorem 1 can be adapted.

Theorem 2. *Let φ be an ERL formula, there exists a finite reduced ON $\mathcal{N}$ such that $\Phi^{gen}_{\mathcal{N}} = \varphi$ iff $CN(\varphi)$ is a reduced ON and $\Phi^{gen}_{CN(\varphi)} = \varphi$.*

With the general semantics, the set of runs cannot be described with the conflict relation only. But since a net $CN(\Phi_{\mathcal{N}})$, built from the formula associated with ON $\mathcal{N}$ has the same causality and conflict relations as $\mathcal{N}$, they accept the same set of general runs.

Finally, there are many possible semantics within the spectrum between maximal and general semantics, in particular time-guarded semantics (Time PNs, arc-timed nets etc.). We believe the analysis shown here will help explore and analyze concurrent behavior also in such complex settings.

D. Acknowledgment

This work was partly supported by the EU FP7 Collaborative project UNIVERSELF, Grant Agreement N^{o}257513 (Call identifier: FP7-ICT-2009-5).

References

[1] S. Haar, "Types of asynchronous diagnosability and the *reveals*-relation in occurrence nets," *IEEE Transactions on Automatic Control*, vol. 55, no. 10, pp. 2310–2320, 2010.

[2] M. Nielsen, G. D. Plotkin, and G. Winskel, "Petri nets, event structures and domains, part I," *Theor. Comput. Sci.*, vol. 13, pp. 85–108, 1981.

[3] J. Engelfriet, "Branching processes of Petri nets," *Acta Inf.*, vol. 28, no. 6, pp. 575–591, 1991.

[4] K. L. McMillan, "Using unfoldings to avoid the state explosion problem in the verification of asynchronous circuits," in *CAV*, ser. LNCS, vol. 663. Springer, 1992, pp. 164–177.

[5] J. Esparza, S. Römer, and W. Vogler, "An improvement of McMillan's unfolding algorithm," *Formal Methods in System Design*, vol. 20, no. 3, pp. 285–310, 2002.

[6] V. Khomenko, "Model checking based on prefixes of Petri net unfoldings," Ph.D. dissertation, School of Computing Science, University of Newcastle upon Tyne, 2003.

[7] W. Vogler, "Fairness and partial order semantics," *Inf. Process. Lett.*, vol. 55, no. 1, pp. 33–39, 1995.

[8] V. Khomenko, M. Koutny, and A. Yakovlev, "Detecting state encoding conflicts in STG unfoldings using SAT," *Fundam. Inf.*, vol. 62, no. 2, pp. 221–241, 2004.

[9] L. Bernardinello, "Synthesis of net systems," in *ICATPN*, ser. LNCS, vol. 691. Springer, 1993, pp. 89–105.

[10] J. Desel and W. Reisig, "The synthesis problem of Petri nets," *Acta Informatica*, vol. 33, pp. 297–315, 1996.

[11] E. Badouel, B. Caillaud, and P. Darondeau, "Distributing finite automata through Petri net synthesis," *Journal on Formal Aspects of Computing*, vol. 13, pp. 447–470, 2002.

[12] J. Carmona, J. Cortadella, M. Kishinevsky, A. Kondratyev, L. Lavagno, and A. Yakovlev, "A symbolic algorithm for the synthesis of bounded Petri nets," in *ICATPN*, ser. LNCS. Springer-Verlag, 2008, vol. 5062, pp. 92–111.

[13] P. Darondeau, "Deriving unbounded petri nets from formal languages," in *CONCUR*, ser. LNCS, vol. 1466. Springer, 1998, pp. 533–548.

[14] R. Bergenthum, J. Desel, R. Lorenz, and S. Mauser, "Synthesis of Petri nets from finite partial languages," *Fundam. Inform.*, vol. 88, no. 4, pp. 437–468, 2008.

[15] A. Ehrenfeucht and G. Rozenberg, "Partial (set) 2-structures. parts I and II," *Acta Inf.*, vol. 27, no. 4, pp. 315–368, 1989.

[16] E. Badouel and P. Darondeau, "Theory of regions," in *Lectures on Petri Nets I: Basic Models*, ser. LNCS. Springer Berlin / Heidelberg, 1998, vol. 1491, pp. 529–586.

[17] P. Gastin and D. Kuske, "Uniform satisfiability problem for local temporal logics over Mazurkiewicz traces," *Information and Computation*, vol. 208, no. 7, pp. 797–816, 2010.

[18] W. Penczek, "Branching time and partial order in temporal logics," in *Time and Logic: A Computational Approach*. UCL Press, 1995, pp. 179–228.