

# On the Model of Computation of Place/Transition Petri Nets

*José Meseguer*<sup>\*</sup>      *Ugo Montanari*<sup>\*\*</sup>  
*Vladimiro Sassone*<sup>\*\*</sup>

<sup>\*</sup>SRI International, Menlo Park, CA 94025, USA

<sup>\*\*</sup>Dipartimento di Informatica, Università di Pisa, Italy

**ABSTRACT.** In the last few years, the semantics of Petri nets has been investigated in several different ways. Apart from the classical “token game”, one can model the behaviour of Petri nets via non-sequential processes, via unfolding constructions, which provide formal relationships between nets and domains, and via algebraic models, which view Petri nets as essentially algebraic theories whose models are monoidal categories.

In this paper we show that these three points of view can be reconciled. More precisely, we introduce the new notion of decorated processes of Petri nets and we show that they induce on nets the same semantics as that of unfolding. In addition, we prove that the decorated processes of a net  $N$  can be axiomatized as the arrows of a symmetric monoidal category which, therefore, provides the aforesaid unification.

## Introduction

*Petri nets*, introduced by C.A. Petri in [18] (see also [21]), are a widely used model of concurrency. This model is attractive from a theoretical point of view because of its simplicity and because of its intrinsically concurrent nature, and has often been used as a semantic basis on which to interpret concurrent languages (see e.g. [27, 17, 26, 5]). Concerning Petri nets themselves, several different semantics have been proposed in the literature. Most of them can be coarsely classified as *process-oriented* semantics, *unfolding* semantics, or *algebraic* semantics, though the latter is not as clearly delimited and is not as widely known as

---

<sup>\*</sup>Supported by Office of Naval Research Contract N00014-92-C-0518, National Science Foundation Grant CCR-9224005, and by the Information Technology Promotion Agency, Japan, as a part of the R & D of Basic Technology for Future Industries “New Models for Software Architecture” sponsored by NEDO (New Energy and Industrial Technology Development Organization).

<sup>\*\*</sup>Partially supported by the EU SCIENCE Programme, Project MASK, and by the Italian National Research Council (CNR), Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo, obiettivo Lambrusco.

the former two classes. Of course, such classes are not at all incomparable, as this paper aims to support. We further discuss these approaches below.

To account for computations involving many different transitions and for the *causal connections* between the “events” which constitute them, the basic notion of computation of Petri nets has been formalized using various notions of *process* [20, 8, 2]. The main criticism raised against process models is that they do not provide a semantics for a net as a whole, but specify only the meaning of single, deterministic computations, while the accurate description of the fine interplay between concurrency and nondeterminism is one of the most valuable features of nets.

Other semantic investigations have capitalized on the *algebraic structure* of Place/Transition (PT) nets, first noticed by Reisig [21] and later exploited by Winskel [29]. The clear advantage of these approaches resides in the fact that they tend to clarify both the structure of the single PT net, so giving insights about their essential properties, and the global structure of the class of all nets. They provide, for example, useful combinators able to describe operations such as parallel and nondeterministic composition of nets [28, 29, 11, 3, 4, 13].

The formal framework which has proved superior for this kind of investigations is *category theory*. The discovery of categories, occurred in the context of algebraic topology in the early forties, emphasized the by now well established conviction that mathematical entities are to be studied in terms of their structure, i.e., in terms of the abstract properties that they enjoy, rather than in terms of their actual elements. Indeed, the theory of categories builds on such conceptual guidelines introducing a new idea: the entities we intend to investigate can be equipped with a notion of *morphism* by means of which all their relevant structural properties can be expressed. (Of course, the actual meanings of “morphism” and “structure” depend on the specific nature of the subject one is considering.) This paradigm is clearly well suited for the study of models of computation, where the entities one considers, i.e., system or behaviour descriptions of some kind, come naturally with an associated notion of “morphism”, e.g., simulations, bisimulations, or similar behavior-based relationships, which encapsulates their real essence. This is in fact also the case of Petri nets whose very structure suggests a notion of morphism which captures the intuitive idea of simulation and, therefore, the idea of behaviour itself. Then, with this understanding of the role of category theory, founding an algebraic theory of Petri nets on categories simply means considering an abstract framework in which *behaviour* is a “first class citizen”. One of the first direct benefits of the use of a categorical framework is that, as a generalization of universal algebra, it provides *universal constructions* which can give fully satisfactory justifications to otherwise ad hoc defined combinators. For example, the parallel and non deterministic compositions of nets discussed above can be understood, respectively, as products and coproducts in the category of nets. An original interpretation of the algebraic structure of PT nets has been proposed in [11], where the theory of *monoidal categories* is exploited to the purpose. Unlike the preceding approaches, [11] yields an algebraic theory of Petri nets in which

notions such as firing sequence, case graph, relationships between net descriptions at different levels of abstraction, duality, and invariants find adequate algebraic/categorical (universal) formulations. Alternative interesting categorical approaches are [3, 4].

In addition to that, since from the formal viewpoint categories are simply *algebraic graphs*, and in particular graphs whose arcs are closed under an operation of sequential composition, it is often the case that the computations of a single behavioural entity, say a Petri net, can be modelled themselves as a category, yielding in this way an axiomatization of its *space of computations*. One may call this use of categories “in the small”, as opposed to their use “in the large” to study the global properties of the entire class of nets as illustrated above. This idea has been exploited in [6], where it is shown that the *commutative processes* [2] of a net  $N$  are isomorphic to the arrows of a symmetric monoidal category  $\mathcal{T}[N]$ . Moreover, [6] introduced the *concatenable processes* of  $N$ —a slight variation of Goltz-Reisig processes [8]—and structured them as the arrows of the symmetric monoidal category  $\mathcal{P}[N]$ . In particular, the distributivity of tensor product and arrow composition in monoidal categories is shown to capture the basic identifications of net computations, thus providing a *model of computation* for Petri nets.

Roughly speaking the *unfolding semantics* consists, as the name indicates, in “unfolding” a net to simple denotational structures such that the identity of every event in their computations is unambiguous. However, *not* every assignment of denotations yields an appropriate semantics for nets. In other words, when defining an unfolding semantics, an integral part of the work is to provide some justification of adequacy of the obtained semantics. Exploiting the categorical framework, it is possible to achieve such a justification implicitly and more satisfactorily than appealing to mere intuition. The idea is to ensure that the denotation assigned to each net enjoys a certain universal property whose role is exactly to guarantee that, for the given target category, the assignment is, informally speaking, “as good as possible”. The theory of categories provides the right notion to express this: the notion of *adjunction*. Thus, one would like to identify an *adjoint functor* assigning a denotation to each PT net and preserving certain compositional properties in the assignment. This is exactly what the present authors—building on Winskel’s work on safe nets [28]—have done in [12, 13] for PT nets (see [7, 9] for related approaches).

In Winskel’s work—which in turn builds on the previous work [15]—the denotation of a *safe* net is a coherent finitary prime algebraic *Scott domain* [25], or *dI-domain* [1]. Winskel shows that there exists a coreflection—a particularly nice form of adjunction—between the category Dom of (coherent) *finitary prime algebraic domains* and the category Safe of *safe Petri nets*. This coreflection factorizes through the chain of coreflections

$$\begin{array}{ccccccc} \underline{\text{Safe}} & \xrightleftharpoons{\mathcal{U}[-]} & \underline{\text{Occ}} & \xrightleftharpoons[\mathcal{N}[-]]{\mathcal{E}[-]} & \underline{\text{PES}} & \xrightleftharpoons[\mathcal{Pr}[-]]{\mathcal{L}[-]} & \underline{\text{Dom}} \end{array}$$

where  $\underline{\text{PES}}$  is the category of *prime event structures* (with binary conflict relation), which is equivalent to  $\underline{\text{Dom}}$ ,  $\underline{\text{Occ}}$  is the category of *occurrence nets* [28] and  $\hookleftarrow$  is the inclusion functor. In [12, 13], such a chain has been extended to a quite general category  $\underline{\text{PTNets}}$  of PT nets by defining the *unfoldings* of PT nets and relating them by means of an *adjunction* to occurrence nets and therefore—exploiting the already existing adjunctions—to prime event structures and finitary prime algebraic domains. Namely, the adjunction between  $\underline{\text{Dom}}$  and  $\underline{\text{PTNets}}$  is the composition of the chain of adjunctions

$$\begin{array}{ccccc}
 \underline{\text{PTNets}} & \xrightleftharpoons[\text{(-)}^+]{\mathcal{U}[-]} & \underline{\text{DecOcc}} & & \\
 & & \mathcal{D}[-] \updownarrow \mathcal{F}[-] & & \\
 & & \underline{\text{Occ}} & \xrightleftharpoons[\mathcal{N}[-]]{\mathcal{E}[-]} & \underline{\text{PES}} \xrightleftharpoons[\mathcal{Pr}[-]]{\mathcal{L}[-]} \underline{\text{Dom}}
 \end{array}$$

where  $\underline{\text{DecOcc}}$  is the “key” category of *decorated occurrence nets*. These are occurrence nets in which places belonging to the post-set of the same transition are partitioned into *families*. In this way, since families are used to relate places corresponding in the unfolding to multiple instances of the same place in the original net, they naturally represent the unfoldings of PT nets and can account for the multiplicities of places in transitions.

We have already mentioned that these three views of net semantics are not mutually exclusive and, in fact, we have discussed how [6] provides a unification of the process-oriented and algebraic views via the categories  $\mathcal{T}[N]$  and  $\mathcal{P}[N]$  modelling, respectively, *commutative* and *concatenable processes*. Concerning the relationships between process and unfolding semantics, in the case of safe nets the question is easily answered by exploiting the existence of a coreflection of  $\underline{\text{Occ}}$  into  $\underline{\text{Safe}}$ , which directly implies the existence of an isomorphism between the processes of  $N$  and the deterministic finite subnets of  $\mathcal{U}[N]$ , i.e., the finite configurations of  $\mathcal{EU}[N]$ . Thus, in this case, the process and unfolding semantics coincide, although it should not be forgotten that the latter has the great merit of collecting together all the processes of  $N$  as a *whole*, thus accounting at the same time for concurrency and nondeterminism.

In this paper we study the relationships between the algebraic paradigm, the process semantics described above, and the unfolding semantics for PT nets given in [12, 13]. We find that, in the context of general PT nets, the latter two notions do not coincide. In particular, the unfolding of a net  $N$  contains information strictly more concrete than the collection of the processes of  $N$ . However, we show that the difference between the two semantics can be axiomatized quite simply. In particular, we introduce a new notion of processes, whose definition is suggested by the idea of families in decorated occurrence nets, and which are therefore called *decorated processes*, and we show that they capture the unfolding semantics, in the precise sense that there is a one-to-one translation between decorated processes of  $N$  and finite configurations of  $\mathcal{EFU}[N]$ .

Then, following the approach of [6], we axiomatize the notion of decorated (concatenable) process in terms of monoidal categories. More precisely, we define an abstract symmetric monoidal category  $\mathcal{DP}[N]$  and we show that its arrows represent *decorated concatenable processes*.

The natural environment for the development of a theory of net processes based on monoidal categories is, as illustrated in [6], a category Petri of *unmarked* nets, i.e., nets without initial markings, whose transitions have finite pre- and post-sets. However, since the unfolding of a net is considered with respect to an initial marking, PTNets and all the categories of nets considered in [12] (and in related works) are categories of *marked* nets whose transitions, because of technical reasons, are forced to have possibly infinite pre- and post-sets and nonempty pre-sets. In order to solve this discrepancy, we simply restrict our attention to the subcategory of PTNets, say MPetri<sup>\*</sup>, consisting of the nets whose transitions have finite pre- and post-sets, i.e., the nets with nonempty pre-sets in Petri equipped with an additional initial marking. Therefore, summing up, our result is that the following diagram commutes up to isomorphism

$$\begin{array}{ccccccc}
 \text{MPetri}^* & \xrightarrow{\mathcal{DP}^*[-]} & \text{MSMonCat} & & & & \\
 \downarrow & & & \searrow \langle -| - \rangle & & & \\
 \text{PTNets} & \xrightarrow{\mathcal{U}[-]} \text{DecOcc} \xrightarrow{\mathcal{F}[-]} \text{Occ} \xrightarrow{\mathcal{E}[-]} \text{PES} & & & \nearrow \mathcal{L}_F[-] & & \text{PreOrd}
 \end{array}$$

where  $\hookrightarrow$  is the inclusion of MPetri<sup>\*</sup> in PTNets, MSMonCat is the category of the “marked” symmetric strict monoidal categories, i.e., symmetric strict monoidal categories  $\underline{\mathcal{C}}$  with a distinguished object  $c \in \underline{\mathcal{C}}$ ,  $\mathcal{DP}^*[-]$  maps the marked net  $(N, u_N)$  to  $(u_N, \mathcal{DP}[N])$ , PreOrd is the category of preorders,  $\langle -| - \rangle$  is the comma category functor  $(c, \underline{\mathcal{C}}) \mapsto \langle c| \underline{\mathcal{C}} \rangle$  (see Definition 3.11), and  $\mathcal{L}_F$  returns the finite configurations of prime event structures ordered by inclusion. It should be stressed that our concern here is at the level of a *single* net, which means that the diagram above is defined only at the object level, i.e., the correspondence we establish is not functorial. Nevertheless, we think that this is an interesting result, since it provides a natural and unified account of the *algebraic*, the *process-oriented*, and the *denotational* views of net semantics. We remark that a similar approach has been followed in [16] in the case of elementary net systems—a particular class of safe nets without self-looping transitions—for unfoldings and non-sequential processes.

To conclude this discussion, we would like to mention that the correspondence of semantics presented here can be lifted smoothly to *infinite computations*. In [24], the present authors show that the symmetric monoidal category  $\mathcal{P}[N]^\omega$  obtained as the completion of  $\mathcal{P}[N]$  by colimits of  $\omega$ -diagrams can be understood as the category of possibly *infinite* concatenable processes of  $N$ . Working analogously, one can see that the arrows of the symmetric strict monoidal category  $\mathcal{DP}[N]^\omega$  are possibly *infinite* decorated concatenable processes. Then, one

can prove the commutativity (up to equivalence) of a diagram analogous to the one above involving all the configurations of  $\mathcal{EFU}[N]$  and the comma category  $\langle u_N \downarrow \mathcal{DP}[N]^\omega \rangle$ . Unfortunately, due to lack of space, we shall not say more about this extension here. The details of this construction can be found in [22].

Concerning the organization of the paper, in Section 1 we recall the basic facts about the algebraic approach to Petri nets as given in [11] and [6]. Then, in Section 2 we give a brief overview of the formal development concerning the unfolding semantics introduced in [12]. In Section 3 we introduce the decorated processes and we illustrate their relationships with the unfolding semantics. Finally, we study the decorated concatenable processes of  $N$  and their axiomatization as the arrows of the symmetric monoidal category  $\mathcal{DP}[N]$ .

Due to the extended abstract nature of this paper, we omit all proofs; the interested reader is referred to [14]. Some of the results presented here appear also in [22].

## 1 Petri Nets and their Processes

In this section we briefly recall the basic definitions about Petri nets (see [21] for a thorough introduction) and their processes.

**NOTATION.** Throughout the paper,  $S^\oplus$  denotes the set of finite multisets on the set  $S$ . We recall that  $S^\oplus$  is a commutative monoid, actually the free commutative monoid on  $S$ , under the operation of multiset union. For  $\mu \in S^\oplus$  we write  $\llbracket \mu \rrbracket$  to indicate the subset of  $S$  consisting of those elements  $s$  such that  $\mu(s) > 0$ . We shall represent a nonempty multiset  $\mu$  as a formal sum  $\bigoplus_{i \in I} n_i s_i$  where  $\{s_i \mid i \in I\} = \llbracket \mu \rrbracket$  and  $n_i = \mu(s_i)$ . The empty multiset, i.e., the unit of the monoid, will be written as 0. Finally, given a finite subset  $S' \subseteq S$ , we shall write  $\bigoplus S'$  for the (multi)set  $\bigoplus_{s \in S'} s$ .

**DEFINITION 1.1** (*Petri Nets and Marked PT Nets*)

A *Place/Transition (PT) Petri net* is a structure  $N = (\partial_N^0, \partial_N^1: T_N \rightarrow S_N^\oplus)$ , where  $T_N$  is a set of transitions,  $S$  is a set of places, and  $\partial_N^0$  and  $\partial_N^1$  are functions such that  $\partial_N^0(t) \neq 0$ .

A *marked PT net* is a pair  $(N, u_N)$ , where  $N$  is a PT net and  $u_N \in S_N^\oplus$  is the initial marking.

This describes a Petri net precisely as a graph whose set of nodes is a free commutative monoid, i.e., the set of *finite multisets* on a given set of *places*. The source and target of an arc, here called a *transition*, are meant to represent, respectively, the *marking* consumed by the transition, i.e., the minimum multiset of tokens which allows the transition to fire, and the marking produced by the firing of the transition. The restriction to nets in which  $\partial_N^0(t) \neq 0$  for each transition  $t$  is due to the fact that such transitions are highly degenerated. In particular, the firing of any number of parallel instances of them is enabled at any marking, and this represents a serious problem for the unfolding semantics.

The formalization of nets as graphs with structure suggests considering graph morphisms which respect such structure as morphisms of nets.

DEFINITION 1.2 (*PT Nets Morphisms*)

A *PT net morphism*  $f: N_0 \rightarrow N_1$  consists of a pair of functions  $\langle f_t, f_p \rangle$ , where  $f_t: T_{N_0} \rightarrow T_{N_1}$  and  $f_p: S_{N_0}^\oplus \rightarrow S_{N_1}^\oplus$  is a monoid homomorphism, such that  $\langle f_t, f_p \rangle$  respects source and target, i.e., it makes the two diagrams below commute.

$$\begin{array}{ccc} T_{N_0} & \xrightarrow{\partial_{N_0}^0} & S_{N_0}^\oplus \\ f_t \downarrow & & \downarrow f_p \\ T_{N_1} & \xrightarrow{\partial_{N_1}^0} & S_{N_1}^\oplus \end{array} \quad \begin{array}{ccc} T_{N_0} & \xrightarrow{\partial_{N_0}^1} & S_{N_0}^\oplus \\ f_t \downarrow & & \downarrow f_p \\ T_{N_1} & \xrightarrow{\partial_{N_1}^1} & S_{N_1}^\oplus \end{array}$$

A *morphism of marked PT nets* from  $N_0$  to  $N_1$  is a *PT net morphism*  $f: N_0 \rightarrow N_1$  which preserves the initial marking, i.e., such that  $f_p(u_{N_0}) = u_{N_1}$ .

NOTATION. To simplify notation we shall almost always omit the subscripts  $t$  and  $p$  which distinguish the components of a morphism  $f$ . The type of the argument will identify which component we are referring to.

Processes provide a causal explanation of net behaviours, which is achieved by decorating the step sequences defined by the “token game” with explicit information about the *causal links* which ruled the firing of the transitions in the sequence. Usually one assumes that such links can be expressed faithfully as a partial order of transitions, the ordering being considered a cause/effect relationship. Thus, roughly speaking, a process of a net  $N$  consists of a partial order built on a multisubset of transitions of  $N$ . The formalization of this gives the following notion of deterministic occurrence net.

NOTATION. In the following, in case of nets without multiplicities, we use the standard notation  $\bullet a$ , for  $a \in S_N$ , to mean the *pre-set* of  $a$ , that is  $\bullet a = \{t \in T_N \mid a \in \llbracket \partial_N^1(t) \rrbracket\}$ . Symmetrically,  $a^\bullet$  indicates  $\{t \in T_N \mid a \in \llbracket \partial_N^0(t) \rrbracket\}$ , the *post-set* of  $a$ . These notations are extended in the obvious way to the case of sets of places. Recall that the terminology pre- and post-set is used also for transitions to indicate, respectively,  $\bullet t = \llbracket \partial_N^0(t) \rrbracket$  and  $t^\bullet = \llbracket \partial_N^1(t) \rrbracket$ . As usual,  $|\cdot|$  indicates the cardinality of sets.

DEFINITION 1.3 (*Occurrence and Process Nets*)

An (nondeterministic) *occurrence net* is a *PT net*  $\Theta$  such that:

i) for all  $t \in T_\Theta$ , for all  $a \in S_\Theta$  one has  $\partial_\Theta^0(t)(a) \leq 1$  and  $\partial_\Theta^1(t)(a) \leq 1$ ;

ii) for all  $a \in S_\Theta$ ,  $|\bullet a| \leq 1$ ;

iii)  $\prec$  is irreflexive, where  $\prec$  is the transitive closure of the relation

$$\prec^1 = \{(a, t) \mid a \in S_\Theta, t \in a^\bullet\} \cup \{(t, a) \mid a \in S_\Theta, t \in \bullet a\};$$

moreover,  $\forall t \in T_\Theta, \{t' \in T_\Theta \mid t' \prec t\}$  is finite;

iv) the binary “conflict” relation  $\#$  on  $T_\Theta \cup S_\Theta$  is irreflexive, where

$$\forall t_1, t_2 \in T_\Theta, t_1 \#_m t_2 \Leftrightarrow \bullet t_1 \cap \bullet t_2 \neq \emptyset \quad \text{and} \quad t_1 \neq t_2,$$

$\forall x, y \in T_\Theta \cup S_\Theta, x \# y \Leftrightarrow \exists t_1, t_2 \in T_\Theta : t_1 \#_m t_2 \text{ and } t_1 \preccurlyeq x \text{ and } t_2 \preccurlyeq y,$   
where  $\preccurlyeq$  is the reflexive closure of  $\prec$ .

Given  $x, y \in T_\Theta \cup S_\Theta$ , we say that  $x$  and  $y$  are concurrent, in symbols  $x \text{ co } y$ , if it is not the case that  $(x \prec y \text{ or } y \prec x \text{ or } x \# y)$ . A set  $X \subseteq T_\Theta \cup S_\Theta$  is concurrent, in symbols  $\text{Co}(X)$ , if  $\forall x, y \in X, x \text{ co } y$  and  $|\{t \in T_\Theta \mid \exists x \in X, t \preccurlyeq x\}| \in \omega$ .

We say that an occurrence net  $\Theta$  is *deterministic* if for all  $a \in S_\Theta$ ,  $|a^\bullet| \leq 1$ . Observe that, in this case, we have  $\# = \emptyset$ . We shall refer to deterministic occurrence nets also as *process nets*.

Thus, in an occurrence nets each place belongs at most to one post-set and, if the net is a process net, at most to one pre-set. This makes the “flow” relation  $\preccurlyeq$  be a pre-order. Thus, requiring  $\prec$  to be irreflexive, which is equivalent to requiring that the net be acyclic, identifies a partial order on the transitions. The constraint about the cardinality of the set of predecessors of a transition is then the fairly intuitive requirement that each transition be *finitely caused*. (See [28] for a discussion in terms of event structures of this issue.)

We stipulate that occurrence nets are to be considered also as *marked* nets whose minimal (wrt.  $\prec$ ) places constitute the initial marking. Observe that this matches exactly with the standard definition, according to which occurrence nets can be marked only by assigning a single token to each of its minimal places. In the following, therefore, we shall use occurrence nets both in contexts in which marked nets are expected and in contexts in which unmarked nets are.

#### DEFINITION 1.4 (*Non-Sequential Processes* [8])

Given a net  $N$ , a *process* of  $N$  is a PT net morphism  $\pi: \Theta \rightarrow N$  which maps places to places (as opposed to morphisms which map places to markings), where  $\Theta$  is a finite process net.

Similarly, a *process* of a marked net  $N$  is a morphism  $\pi: \Theta \rightarrow N$  of marked PT nets which maps places to places, for a finite process net  $\Theta$ .

For the purpose of defining processes at the right level of abstraction, we need to make some identifications among process nets. Of course, we shall consider as identical process nets which are isomorphic and, consequently, we shall make no distinction between two processes  $\pi: \Theta \rightarrow N$  and  $\pi': \Theta' \rightarrow N$  for which there exists an isomorphism  $\varphi: \Theta \rightarrow \Theta'$  such that  $\pi' \circ \varphi = \pi$ . Observe that the particular form of  $\pi$  is relevant, since we certainly want process morphisms to be total and to map a single component of the process net to a single component of  $N$ . Otherwise said, process morphisms are nothing but *labellings* of  $\Theta$  with an appropriate element of  $N$ . Moreover, as usual, in the case of marked nets, we want to consider only processes whose source is the initial marking.

Inspired by the current trends in the development of the theory of computation, one would certainly like to describe the processes of a net  $N$  as an algebra whose operations model a minimal set of combinators on processes which capture the essence of concurrency. Clearly, in the present case the core of such



an algebra must consist of the operations of *sequential* and *parallel* composition of processes. The problem which arises immediately is that non-sequential processes cannot be concatenated when multiplicities are present: in order to support such an operation one must *disambiguate* the identity of all the tokens in the multisets source and target of processes. In other words, one must recognize that process concatenation has to do with tokens rather than with places. This is the approach followed in [6], which led to the introduction of the *concatenable processes* of  $N$ . These are, as already sketched above, non-sequential processes enriched by total orderings of the minimal and maximal places carrying the same label. Then, exploiting the additional information, it is easy to define an operation of concatenation of such processes, and thus to organize them as the arrows of a category  $\mathcal{CP}[N]$ . In particular, since concatenable processes also admit an operation of parallel composition,  $\mathcal{CP}[N]$  is a symmetric monoidal category. In addition, [6] shows that  $\mathcal{CP}[N]$  can be axiomatized by means of an abstract symmetric monoidal category  $\mathcal{P}[N]$  (see also [23]).

## 2 Unfolding Place/Transition Nets

In this section we sketch the basic notions concerning the unfolding of PT Petri nets as defined in [12, 13]. In order to keep the exposition of the background material as short as possible, we limit ourselves to the definitions of the object components of the functors  $\mathcal{U}[-]$ ,  $\mathcal{F}[-]$ ,  $\mathcal{E}[-]$  and  $\mathcal{L}[-]$ . In particular, we shall not introduce explicitly the categories involved. The reader interested in the details is referred to [12, 28]. A complete survey of the topic is also given in [22].

As a first step, we define *decorated occurrence nets*, a type of occurrence nets in which places are grouped into families. They allow a convenient treatment of multiplicity issues in the unfolding of PT nets. We shall use  $[n]$  to denote the segment  $\{1, \dots, n\}$  of  $\omega$ .

**DEFINITION 2.1** (*Decorated Occurrence Nets [12]*)

A decorated occurrence net is an occurrence net  $\Theta$  such that:

- i)  $S_\Theta$  is of the form  $\bigcup_{a \in A_\Theta} \{a\} \times [n_a]$ , for some set  $A_\Theta$ , where the set  $\{a\} \times [n_a]$  is called the family of  $a$ . We will use  $a^F$  to denote the family of  $a$  regarded as a multiset;
- ii)  $\forall a \in A_\Theta, \forall x, y \in \{a\} \times [n_a], \bullet x = \bullet y$ .

A family is thus a collection of finitely many places with the same pre-set, and a decorated occurrence net is an occurrence net where each place belongs to exactly one family. Families, and therefore decorated occurrence nets, are capable of describing *relationships* between places by grouping them together. We will use families to relate places which are *instances* of the same place obtained in a process of unfolding.

Next, we define an unfolding procedure which maps marked PT nets to decorated occurrence nets.

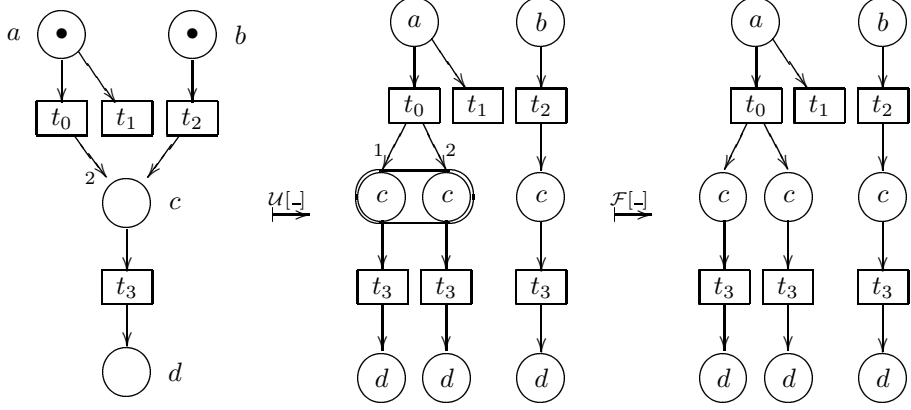


Figure 1: A net  $N$ , its unfolding  $\mathcal{U}[N]$ , and  $\mathcal{FU}[N]$

DEFINITION 2.2 (*PT Nets Unfoldings:  $\mathcal{U}[\cdot]$  [12]*)

Let  $N = (\partial_N^0, \partial_N^1: T_N \rightarrow S_N^\oplus, u_N)$  be a marked net. We define the decorated occurrence net  $\mathcal{U}[N]$  to be  $(\partial^0, \partial^1: T \rightarrow S^\oplus)$ , where  $T$ ,  $S$  and  $\partial^0$  are generated inductively by the following inference rules.

$$\begin{array}{c}
 \frac{u_N(b) = n}{\left\{ (\emptyset, b) \right\} \times [n] \subseteq S} \\
 \\
 \frac{B = \left\{ \left( (\epsilon_j, b_j), i_j \right) \mid j \in J \right\} \subseteq S, \text{ Co}(B), t \in T_N, \partial_N^0(t) = \bigoplus_{j \in J} b_j}{(B, t) \in T \text{ and } \partial^0(B, t) = \bigoplus B} \\
 \\
 \frac{x = (B, t) \in T, \partial_N^1(t)(b) = n}{\left\{ (\{x\}, b) \right\} \times [n] \subseteq S}
 \end{array}$$

and for  $x \in T$ ,  $\partial^1(x) = \bigoplus_{b,i} \left( (\{x\}, b), i \right)$ .

Informally speaking, the definition above can be explained as follows, where we use  $\mathcal{U}[N]^{(n)}$ ,  $n \in \omega$ , to denote the  $n$ -th approximation of  $\mathcal{U}[N]$ , i.e., the subnet of  $\mathcal{U}[N]$  consisting of the elements at depth not greater than  $n$ . The net  $\mathcal{U}[N]^{(0)}$  is obtained by exploding in families the initial marking of  $N$ , and  $\mathcal{U}[N]^{(n+1)}$  is obtained, inductively, by generating a new transition for each possible subset of concurrent places of  $\mathcal{U}[N]^{(n)}$  whose corresponding multiset of places of  $N$  constitutes the source of some transition  $t$  of  $N$ ; the target of  $t$  is also exploded in families which are added to  $\mathcal{U}[N]^{(n+1)}$ . As a consequence, the transitions of the  $n$ -th approximant net are instances of transitions of  $N$ , in the precise sense that each of them corresponds to a unique occurrence of a transition of  $N$  in one of its step sequences of length at most  $n$ .

There is an obvious *forgetful* functor from decorated occurrence nets to occurrence nets which forgets about the structure of families. It allows us to drop the additional structure of decorated occurrence nets and to bring the unfolding of PT nets into Occ. Moreover, exploiting Winskel's coreflections in [28], we obtain an explanation of the causal behaviour of nets in PES and in Dom as illustrated in the introduction.

DEFINITION 2.3 ( $\mathcal{F}[\cdot]$ : from DecOcc to Occ [12])

Given a decorated occurrence net  $\Theta$  define  $\mathcal{F}[\Theta]$  to be the occurrence net underlying  $\Theta$ .

Figure 1 shows a simple example of unfolding of PT nets. To make explicit the nature of the elements of  $\mathcal{U}[N]$  and  $\mathcal{FU}[N]$ , in the picture we label them with the corresponding element  $a, b, \dots, t_3$  of  $N$ . In particular, the places of the unfolding labelled by  $a$  and  $b$  are respectively  $(\emptyset, a)$  and  $(\emptyset, b)$ , the transitions labelled by  $t_0$  and  $t_2$  are  $\bar{t}_0 = (\{(\emptyset, a)\}, t_0)$  and  $\bar{t}_2 = (\{(\emptyset, b)\}, t_2)$ , and thus the three instances of  $c$  are  $((\{\bar{t}_0\}, c), 1)$ ,  $((\{\bar{t}_0\}, c), 2)$  and  $((\{\bar{t}_2\}, c), 1)$ . A family is represented by enclosing its elements into an oval. The numbers which label the outgoing arcs from  $\bar{t}_0$  take into account the ordering of the elements in the family  $(\{\bar{t}_0\}, c)^F$ ; since  $\mathcal{U}[N]$  is an occurrence net, no confusion is possible with arc multiplicities. Families of cardinality one are not explicitly indicated. We call  $\mathcal{U}[N]$  and  $\mathcal{FU}[N]$  respectively the unfolding of  $N$  in DecOcc and in Occ. However, in the following we shall avoid explicit reference to DecOcc and Occ.

Prime event structures [15, 28] are the simplest event based model of concurrency. They consist of a set of events, intended as indivisible *quanta* of computation, which are related to each other by two binary relation: *causality*, modelled by a partial order relation  $\leq$ , and *conflict*, modelled by an irreflexive, symmetric, and hereditary relation  $\#$ .

DEFINITION 2.4 (*Prime Event Structures*)

A prime event structure is a structure  $E = (E, \#, \leq)$  consisting of a set of events  $E$  partially ordered by  $\leq$ , and a symmetric, irreflexive relation  $\# \subseteq E \times E$ , the conflict relation, such that

$$\begin{aligned} \{e' \in E \mid e' \leq e\} \text{ is finite for each } e \in E \\ e \# e' \leq e'' \text{ implies } e \# e'' \text{ for each } e, e', e'' \in E. \end{aligned}$$

The computational intuition behind event structures is really simple: an event  $e$  can occur when all its *causes* have occurred and no event that is in conflict with the given event has already occurred. This is formalized by the following notion of *configuration*.

DEFINITION 2.5 (*Configurations*)

Given a prime event structure  $(E, \#, \leq)$ , define its configurations to be those subsets  $x \subseteq E$  which are

$$\begin{aligned} \text{Conflict Free: } \forall e_1, e_2 \in x, \text{ not}(e_1 \# e_2) \\ \text{Left Closed: } \forall e \in x \forall e' \leq e, e' \in x \end{aligned}$$

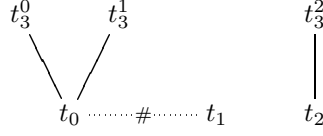


Figure 2: The event structure  $\mathcal{EFU}[N]$  for the net in Figure 1

Let  $\mathcal{L}(E)$  denote the set of configurations of the prime event structure  $E$  and  $\mathcal{L}_F(E)$  the set of finite configurations of  $E$ .

The following definition recalls how to translate occurrence nets into prime event structures. An example of this translation is shown in Figure 2, where, using the standard graphical representation of event structures,  $\leq$  is indicated by (bottom-up) solid lines and  $\#$  by a dotted line; we use superscripts to distinguish between the three instances of  $t_3$  in  $\mathcal{FU}[N]$ .

**DEFINITION 2.6** ( $\mathcal{E}[\cdot]$ : from Occ to PES [28])

Let  $\Theta$  be an occurrence net. Then,  $\mathcal{E}[\Theta]$  is the event structure  $(T_\Theta, \preceq, \#)$ , where  $\preceq$  and  $\#$  are the restrictions to the set of transitions of  $\Theta$  of, respectively, the flow ordering and the conflict relation implicitly defined by  $\Theta$ .

Finitary prime algebraic domains or dI-domains—introduced by G. Berry while studying sequentiality of functions [1]—are particular Scott’s domains which are distributive and in which each finite element is preceded only by a finite number of elements of the domain. Here we are interested in their “coherent” version, i.e., in the version in which the underlying partial order is pairwise complete.

**DEFINITION 2.7** (Finitary (Coherent) Prime Algebraic Domains)

Let  $(D, \sqsubseteq)$  be a partial order. Recall that a set  $X \subseteq D$  is directed if all the pairs  $x, y \in X$  have an upper bound in  $X$ , is compatible if there exists  $d \in D$  such that  $x \sqsubseteq d$  for all  $x \in X$  and is pairwise compatible if  $\{x, y\}$  is compatible for all  $x, y \in X$ . We say that  $D$  is a (coherent) domain if it is pairwise complete, i.e., if for all pairwise compatible  $X \subseteq D$  the least upper bound  $\bigsqcup X$  of  $X$  exists.

A complete prime of  $D$  is an element  $p \in D$  such that, for any compatible  $X \subseteq D$ , if  $p \sqsubseteq \bigsqcup X$ , then there exists  $x \in X$  such that  $p \sqsubseteq x$ . We say that a domain  $D$  is prime algebraic if for all  $d \in D$  we have  $d = \bigsqcup \{p \sqsubseteq d \mid p \text{ is a complete prime}\}$ .

Moreover, an element  $e \in D$  is finite if for any directed  $S \subseteq D$ , if  $e \sqsubseteq \bigsqcup S$ , then there exists  $s \in S$  such that  $e \sqsubseteq s$ . We say that  $D$  is finitary if for all finite elements  $e \in D$ ,  $|\{d \in D \mid d \sqsubseteq e\}| \in \omega$ .

Finitary prime algebraic domains can be equipped with a notion of morphism in such a way that the category Dom so obtained is equivalent to PES (see [28]). We conclude this section by recalling the object component of the equivalence functor  $\mathcal{L}[\cdot]: \text{PES} \rightarrow \text{Dom}$ . An example is provided by Figure 3.

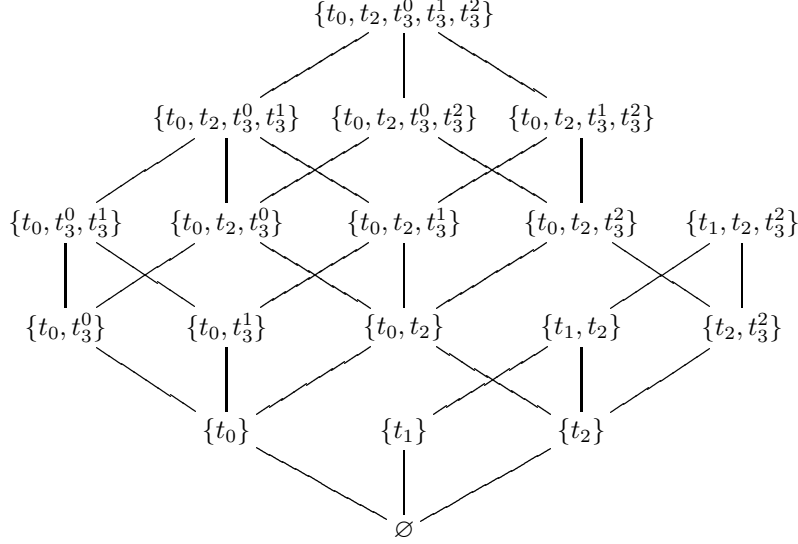


Figure 3: The Hasse diagram of the domain  $\mathcal{LEFU}[N]$  for the net in Figure 1

PROPOSITION 2.8 ( $\mathcal{L}[\cdot]$ : from PES to Dom [28])

Let  $E$  be a prime event structure. Then,  $\mathcal{L}(E) = (\mathcal{L}(E), \subseteq)$ , i.e., the set of configurations of  $E$  ordered by inclusion is a finitary (coherent) prime algebraic domain.

### 3 Process vs. Unfolding Semantics for Nets

The semantics obtained via the unfolding yields an explanation of the behaviour of nets in terms of event structures, that is, in terms of domains. Domains can be unambiguously thought of as partial orderings of computations, where a computation is represented by a configuration, which, in our context, is a “downward” closed, conflict free set of occurrences of transitions. On the other hand, processes are by definition left closed and conflict free (multi)sets of transitions. Moreover, the processes from a given initial marking are naturally organized in a preorder-like fashion via a comma category construction which formalizes the usual notion of prefix ordering of processes. The question which therefore arises spontaneously concerns the relationships between these two notions; this is the question addressed in this section.

It is worth noticing that in the case of safe nets the question is readily answered exploiting Winskel’s coreflection  $\langle \hookrightarrow, \mathcal{U}[\cdot] \rangle: \underline{\text{Occ}} \rightarrow \underline{\text{Safe}}$ . In fact, by definition an adjunction  $\langle F, G \rangle: \underline{\mathcal{C}} \rightarrow \underline{\mathcal{D}}$  determines an isomorphism between arrows of the kind  $F(c) \rightarrow d$  in  $\underline{\mathcal{D}}$  and arrows of the kind  $c \rightarrow G(d)$  in  $\underline{\mathcal{C}}$ . Then,

in the case of safe nets, we have a one-to-one correspondence

$$\pi: \Theta \rightarrow N \iff \pi': \Theta \rightarrow \mathcal{U}[N]$$

for each safe net  $N$  and each occurrence net  $\Theta$ . Therefore, since such correspondence is easily seen to map processes to processes, in this special case, the correspondence between process and unfolding semantics of  $N$  is very tidy: they are the same notion in the precise sense that there is an isomorphism between the processes of  $N$  and the processes of  $\mathcal{U}[N]$ , i.e., the deterministic finite subnets of the unfolding of  $N$ , i.e., the finite configurations of  $\mathcal{EU}[N]$ .

In our context, however, we have that the unfolding of  $N$  is strictly more concrete than the processes of  $N$ . For example, consider again the net  $N$  and its unfolding  $\mathcal{FU}[N]$  shown in Figure 1. Clearly, there is a unique process of  $N$  in which  $t_0, t_2$  and a single instance of  $t_3$  caused by  $t_0$  has occurred. Nevertheless, there are two deterministic subnets of  $\mathcal{FU}[N]$  which correspond to such process, namely those obtained by choosing respectively the left and the right instance of  $t_3$  below  $t_0$ . It is worth noticing that such subnets are isomorphic and that this is not a fortunate case, since it is easy to show that two finite deterministic subnets of  $\mathcal{FU}[N]$  correspond to the same process of  $N$  if and only if they are isomorphic via an isomorphism which sends instances of an element of  $N$  to instances of the same element. More interestingly, the results of this paper will prove that this is the exact relationship between the two semantics of  $N$ : the unfolding contains several copies of the same process which, as illustrated in [12, 13], are needed to provide a fully *causal* explanation of the behaviour of  $N$ , i.e., to obtain an occurrence net whose transitions represent exactly the instances of the transitions of  $N$  in all the possible causal contexts and which can therefore account for concurrent multiple instances of the same element of  $N$ , that is for *autoconcurrency*. More precisely, we shall see that the finite deterministic subnets of the unfolding of  $N$  can be characterized by appropriately *decorating* the processes of  $N$ , which shows directly that the difference between process and unfolding semantics of  $N$  is due only to the replication of data needed in the latter.

Of course, the appropriate decoration of processes is immediately suggested by the notion of family in decorated occurrence nets: a *decorated process* is simply a process whose underlying process net is a decorated occurrence net. This yields a *process-oriented* account of the unfolding construction. In addition, we shall also identify an abstract *symmetric strict monoidal category*  $\mathcal{DP}[N]$  whose arrows axiomatize the decorated *concatenable* processes of  $N$ , and which therefore provides both the *algebraic* and the *process-oriented* account of the unfolding construction. In particular, for each marked PT net  $(N, u_N)$  we have  $\mathcal{L}_F\mathcal{EFU}[(N, u_N)] \cong \langle u_N \downarrow \mathcal{DP}[N] \rangle$ , where the role of the comma category construction is to consider only the decorated concatenable processes from the initial marking  $u_N$ . Therefore, decorated (*deterministic*) occurrence nets, which at first seem to be just a convenient technical solution to establish the adjunction from PT nets to occurrence nets, provide *both* the process and the algebraic counterpart of the unfolding semantics.

The following proposition remarks the intuitive relationship between finite configurations and processes in the unfolding semantics.

**PROPOSITION 3.1**

*Let  $N$  be a marked net. There is an isomorphism between the set of finite configurations of  $\mathcal{EFU}[N]$  and the set of (marked) processes of  $\mathcal{FU}[N]$ .*

Our next task is to characterize the processes of  $\mathcal{FU}[N]$  in terms of processes of  $N$ . We shall do it by means of the following notion of decorated process.

**DEFINITION 3.2 ( $f$ -indexed orderings)**

*Given sets  $A$  and  $B$  together with a function  $f: A \rightarrow B$ , an  $f$ -indexed ordering of  $A$  is a family  $\{\ell_b \mid b \in B\}$  of bijections  $\ell_b: f^{-1}(b) \rightarrow \{1, \dots, |f^{-1}(b)|\}$ , with  $f^{-1}(b)$  being as usual the set  $\{a \in A \mid f(a) = b\}$ .*

Therefore, an  $f$ -indexed ordering of  $A$  is a family of total orderings, one for each of the partitions of  $A$  induced by  $f$ .

**DEFINITION 3.3 (Decorated Processes)**

*Let  $N$  be a marked net. A decorated process of  $N$  is a triple  $DP = (\pi, \ell, \tau)$*

- $\pi: \Theta \rightarrow N$  is a (marked) process of  $N$ ;
- $\ell$  is a  $\pi$ -indexed ordering of  $\min(\Theta)$ , the minimal (wrt.  $\preceq$ ) places of  $\Theta$ ;
- $\tau$  is a family  $\{\tau(t)\}$  indexed by the transitions  $t$  of  $\Theta$ , where each  $\tau(t)$  is a  $\pi$ -indexed ordering of the post-set of  $t$  in  $\Theta$ .

*The decorated processes  $(\pi: \Theta \rightarrow N, \ell, \tau)$  and  $(\pi': \Theta' \rightarrow N, \ell', \tau')$  are isomorphic, and then identified, if their underlying processes are isomorphic via an isomorphism  $\varphi$  which respects all the orderings, i.e.,  $\ell'_{\pi'(\varphi(a))}(\varphi(a)) = \ell_{\pi(a)}(a)$  for all  $a \in \min(\Theta)$ , and  $\tau'(\varphi(t))_{\pi'(\varphi(a))}(\varphi(a)) = \tau(t)_{\pi(a)}(a)$  for all  $t \in T_\Theta$  and for all  $a \in t^\bullet$ .*

*We say that  $(\pi: \Theta \rightarrow N, \ell, \tau) \leq (\pi': \Theta' \rightarrow N, \ell', \tau')$  if there exists  $\varphi: \Theta \rightarrow \Theta'$  which preserves all the orderings and such that  $\pi = \pi' \circ \varphi$ . Since we identify isomorphic processes, the set of decorated processes is partially ordered by  $\leq$ . We shall write  $DP[N]$  to indicate such ordering.*

Figure 4 shows the two decorated processes of the net  $N$  in Figure 1 corresponding to the (unique) process of  $N$  in which  $t_0$ ,  $t_2$  and an instance of  $t_3$  caused by  $t_0$  have occurred. In the pictures, we represent a process  $\pi: \Theta \rightarrow N$  by drawing  $\Theta$  and labelling its element  $x$  by  $\pi(x)$ . Observe that Figure 4 also gives a hint about the announced correspondence.

**PROPOSITION 3.4**

*The set of decorated processes of  $N$  is isomorphic to the set of (marked) processes of  $\mathcal{FU}[N]$ .*

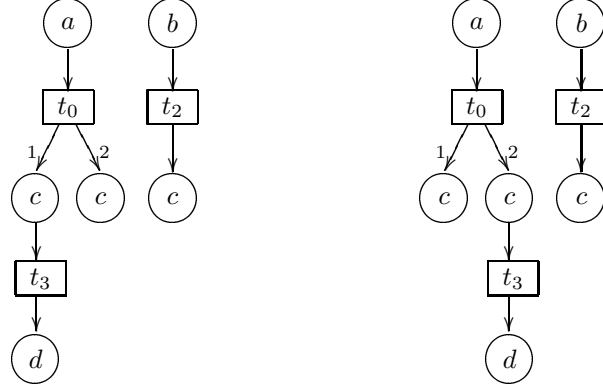


Figure 4: Two decorated processes of the net in Figure 1

Since  $\mathcal{FU}[N]$  is obtained from a decorated occurrence net via a forgetful functor, a process  $\pi$  of  $\mathcal{FU}[N]$  still contains information about the families of  $\mathcal{U}[N]$ . Informally speaking, the proof of the previous proposition consists of showing that it is possible to move such information from  $\pi$  to  $\ell$  and  $\tau$  of a decorated process of  $N$  and back in  $\pi$  from  $\ell$  and  $\tau$ , so obtaining an isomorphism. The correspondence above can be easily lifted to the partial orders of decorated processes and finite configurations of  $\mathcal{EFU}[N]$ .

PROPOSITION 3.5

$DP[N]$  is isomorphic to  $\mathcal{L}_F\mathcal{EFU}[N]$ .

Exploiting further the idea of decorated processes, the same conceptual step which led from non-sequential processes to concatenable processes suggests the following definition.

DEFINITION 3.6 (*Decorated Concatenable Processes*)

A decorated concatenable process of the (unmarked) net  $N$ , is a quadruple  $(\pi, \ell, \tau, L)$  where  $\pi$  is a process of  $N$ ,  $\ell$  and  $L$  are, respectively,  $\pi$ -indexed orderings of  $\min(\Theta)$  and  $\max(\Theta)$ , i.e., the minimal and the maximal places of  $\Theta$ , and  $\tau$  is a family  $\{\tau(t)\}$  indexed by the transitions  $t$  of  $\Theta$ , where each  $\tau(t)$  is a  $\pi$ -indexed ordering of the post-set of  $t$  in  $\Theta$ .

An isomorphism of decorated concatenable processes is an isomorphism of the underlying processes which, in addition, preserves all the orderings given by  $\ell$ ,  $\tau$  and  $L$ .

So, a decorated concatenable process  $(\pi, \ell, \tau, L)$  is a concatenable process  $(\pi, \ell, L)$  where the post-sets of all transitions are  $\pi$ -indexed ordered. Observe that  $(\pi, \ell, \tau)$  is different from a decorated process since  $\pi$  is *unmarked*. It follows from the definitions that a place in  $\max(\Theta)$  has in every case a double ordering: derived from  $L$  and  $\ell$  if it is also minimal, and derived from  $L$  and  $\tau$  otherwise.



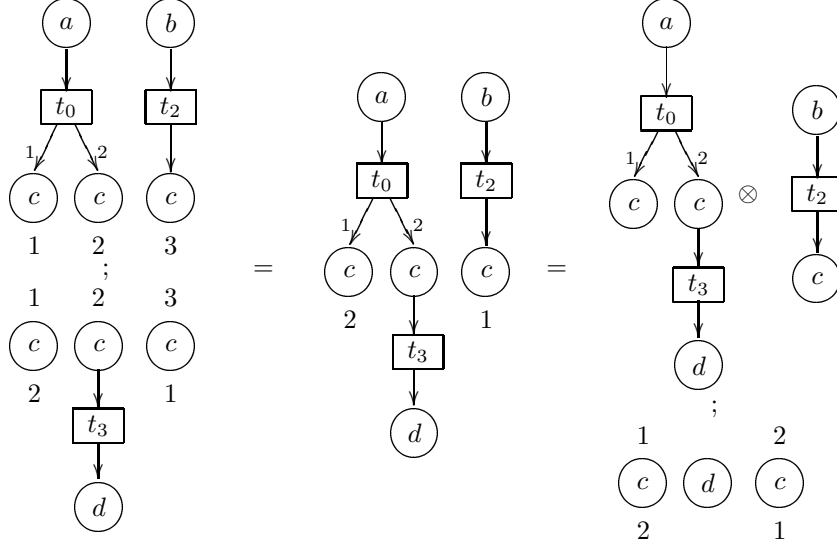


Figure 5: An example of the algebra of decorated concatenable processes

It is clearly possible to define an operation of concatenation of decorated concatenable processes. We can associate a source and a target in  $S_N^\oplus$  to any concatenable process  $DCP$ , namely by taking the image through  $\pi$  of, respectively,  $\min(\Theta)$  and  $\max(\Theta)$ , where  $\Theta$  is the underlying process net of  $DCP$ . Then, the concatenation of  $DCP_0: u \rightarrow v$  and  $DCP_1: v \rightarrow w$  is defined in the obvious way exploiting the informations given by the labellings in order to merge the maximal places of the process nets underlying  $DCP_0$  with the minimal places of the process net underlying  $DCP_1$ . (See also Figure 5, where the elements of the nets are labelled according to  $\pi$ ,  $\ell$ , and  $L$ ,  $\tau(t)$  is represented by decorating the arcs outgoing from  $t$  and all the trivial orderings are omitted.) Therefore, we can consider the category  $\mathcal{DCP}[N]$  whose objects are the finite multisets on  $S_N$  and whose arrows are the decorated concatenable processes.

**PROPOSITION 3.7**

*Under the above defined operation of sequential composition,  $\mathcal{DCP}[N]$  is a category with identities those decorated concatenable processes consisting only of places, which therefore are both minimal and maximal, and such that  $\ell = L$ .*

Decorated concatenable processes admit also a tensor operation  $\otimes$  which represents the parallel composition of processes. Given  $DCP_0: u \rightarrow v$  and  $DCP_1: u' \rightarrow v'$ ,  $DCP_0 \otimes DCP_1: u \oplus u' \rightarrow v \oplus v'$  is the decorated concatenable process which may be graphically represented by putting side by side, from left to right, the graphical representations of  $DCP_0$  and  $DCP_1$  and reorganizing

the labellings appropriately as shown in Figure 5. It is easy to see that the concatenable processes consisting only of places are the *symmetries* [10] which make  $\mathcal{DCP}[N]$  into a *symmetric strict monoidal category*.

PROPOSITION 3.8

$\mathcal{DCP}[N]$  is a *symmetric strict monoidal category*.

Recalling that decorated concatenable processes are a refinement of concatenable processes and that the concatenable processes of  $N$  correspond to the arrows of a category  $\mathcal{P}[N]$ , we are led to the following definition of the symmetric monoidal category  $\mathcal{DP}[N]$  which captures the *algebraic essence* of decorated (concatenable) processes, and thus of the unfolding construction. It is worth remarking that the definition of  $\mathcal{DP}[N]$  can be obtained simply by dropping one axiom from the definition of  $\mathcal{P}[N]$ . This shows that the difference between processes and decorated processes is simply axiomatizable in terms of monoidal categories.

DEFINITION 3.9 (*The category  $\mathcal{DP}[N]$* )

Let  $N$  be a PT net. Then  $\mathcal{DP}[N]$  is the monoidal quotient of the free symmetric strict monoidal category on  $N$  modulo the axioms

$$\begin{aligned} c_{a,b} &= id_{a \oplus b} & \text{if } a, b \in S_N \text{ and } a \neq b \\ s; t &= t & \text{if } t \in T_N \text{ and } s \text{ is a symmetry.} \end{aligned}$$

where  $c$  is the symmetry natural isomorphism. Explicitly, the category  $\mathcal{DP}[N]$  is the category whose objects are the elements of  $S_N^\oplus$  and whose arrows are generated by the inference rules

$$\begin{array}{c} \frac{u \in S_N^\oplus}{id_u: u \rightarrow u \text{ in } \mathcal{DP}[N]} \quad \frac{u, v \text{ in } S_N^\oplus}{c_{u,v}: u \oplus v \rightarrow u \oplus v \text{ in } \mathcal{DP}[N]} \quad \frac{t: u \rightarrow v \text{ in } T_N}{t: u \rightarrow v \text{ in } \mathcal{DP}[N]} \\ \frac{\alpha: u \rightarrow v \text{ and } \beta: u' \rightarrow v' \text{ in } \mathcal{DP}[N]}{\alpha \otimes \beta: u \oplus u' \rightarrow v \oplus v' \text{ in } \mathcal{DP}[N]} \quad \frac{\alpha: u \rightarrow v \text{ and } \beta: v \rightarrow w \text{ in } \mathcal{DP}[N]}{\alpha; \beta: u \rightarrow w \text{ in } \mathcal{DP}[N]} \end{array}$$

modulo the axioms expressing that  $\mathcal{DP}[N]$  is a strict monoidal category, namely,

$$\begin{aligned} \alpha; id_v &= \alpha = id_u; \alpha & \text{and} & \quad (\alpha; \beta); \delta = \alpha; (\beta; \delta), \\ (\alpha \otimes \beta) \otimes \delta &= \alpha \otimes (\beta \otimes \delta) & \text{and} & \quad id_0 \otimes \alpha = \alpha = \alpha \otimes id_0, \\ id_u \otimes id_v &= id_{u \oplus v} & \text{and} & \quad (\alpha \otimes \alpha'); (\beta \otimes \beta') = (\alpha; \beta) \otimes (\alpha'; \beta'), \end{aligned} \tag{1}$$

the latter whenever the lefthand term is defined, the following axioms expressing that  $\mathcal{DP}[N]$  is symmetric with symmetry isomorphism  $c$

$$\begin{aligned} c_{u,v \oplus w} &= (c_{u,v} \otimes id_w); (id_v \otimes c_{u,w}), \\ c_{u,u'}; (\beta \otimes \alpha) &= (\alpha \otimes \beta); c_{v,v'} \quad \text{for } \alpha: u \rightarrow v, \beta: u' \rightarrow v', \\ c_{u,v}; c_{v,u} &= id_{u \oplus v}, \end{aligned} \tag{2}$$

and the following axioms

$$\begin{aligned} c_{a,b} &= id_{a \oplus b} && \text{if } a, b \in S_N \text{ and } a \neq b \\ (id_u \otimes c_{a,a} \otimes id_v); t &= t && \text{if } t \in T_N. \end{aligned} \quad (3)$$

It is worthwhile to remark that in the definition above axioms (1) and (2) define  $\mathcal{F}(N)$ , the free symmetric strict monoidal category on  $N$ . Observe that, exploiting the coherence axiom, i.e., the first of (2), a symmetry in  $\mathcal{F}(N)$  can always be written as a composition of symmetries of the kind  $(id_u \otimes c_{a,b} \otimes id_v)$  for  $a, b \in S_N$ . Then, since we have  $c_{a,b} = id_{a \oplus b}$  if  $a \neq b$ , the axiom  $s; t = t$  takes the particular form stated in (3).

For  $\mathcal{DP}[N]$  and  $\mathcal{DCP}[N]$  we have the following result which matches the analogous one for concatenable processes [6].

PROPOSITION 3.10

$\mathcal{DCP}[N]$  and  $\mathcal{DP}[N]$  are isomorphic.

*Proof.* (Sketch.) Consider the following mapping  $F$  from the arrows of  $\mathcal{DP}[N]$  to decorated concatenable processes.

- A transition  $t$  of  $N$  is mapped to the decorated concatenable processes with a unique transition and two layers of places: the minimal, in one-to-one correspondence with  $\partial_N^0(t)$ , and the maximal, in one-to-one correspondence with  $\partial_N^1(t)$ . The decoration, of course, consists in taking  $\tau(t) = L$ .
- A symmetry  $c_{u,v}$ , for  $u = n_1 a_1 \oplus \dots \oplus n_k a_k$  and  $v = m_1 b_1 \oplus \dots \oplus m_h b_h$  is mapped to the concatenable process having as many places as elements in the multiset  $u \oplus v$  mapped by  $\pi$  to the corresponding places of  $N$  and such that  $L_{a_i}(x) = v(a_i) + \ell_{a_i}(x)$  and  $\ell_{b_i}(x) = L_{b_i}(x) - u(b_i)$ .
- $F$  is extended inductively to a generic term  $\alpha$  of  $\mathcal{DP}[N]$ , i.e.,  $\alpha_0 \otimes \alpha_1$  is mapped to  $F(\alpha_0) \otimes F(\alpha_1)$  and  $\alpha_0 ; \alpha_1$  to  $F(\alpha_0); F(\alpha_1)$ .

Then, defining  $F$  to be the identity on the objects gives the required isomorphism  $F: \mathcal{DP}[N] \cong \mathcal{DCP}[N]$ .  $\checkmark$

We conclude the section by getting back to the diagram discussed in the introduction. We recall the following simple notion from category theory.

DEFINITION 3.11 (*Comma Categories*)

Let  $\underline{\mathcal{C}}$  be a category and  $c$  an object of  $\underline{\mathcal{C}}$ . Then, the comma category  $\langle c \downarrow \underline{\mathcal{C}} \rangle$ , also called the category of elements under  $c$ , is the category whose objects are the arrows  $f: c \rightarrow c'$  of  $\underline{\mathcal{C}}$  and whose arrows  $h: (f: c \rightarrow c') \rightarrow (g: c \rightarrow c'')$  are the commutative diagrams

$$\begin{array}{ccc} & c & \\ f \swarrow & & \searrow g \\ c' & \xrightarrow{h} & c'' \end{array}$$

Identities and arrow composition are inherited in the obvious way from  $\underline{\mathcal{C}}$ .

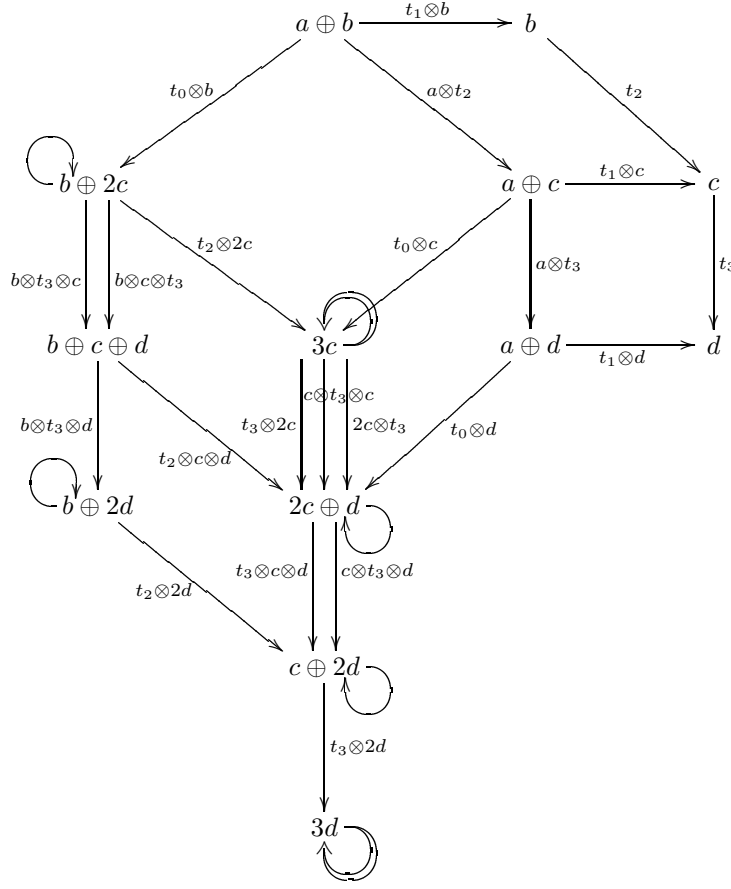


Figure 6: Some of the arrows with source  $a \oplus b$  in  $\mathcal{DP}[N]$  for the net of Figure 1

The next proposition essentially shows that the canonical partial order associated to  $\langle u_N \downarrow N \rangle$  is  $DP[(N, u_N)]$ , and concludes our exposition. The proof follows easily from Proposition 3.10, and the intuition behind it can be grasped from Figure 6, where the self-looping arrows represent the non-identity symmetries. We warn the reader that not all the symmetries are shown in the picture; this is the meaning of the double arrows which stand for several of them.

**PROPOSITION 3.12**

For any marked PT net  $(N, u_N)$ ,

$$\langle u_N \downarrow \mathcal{DP}[N] \rangle \cong DP[(N, u_N)] \cong \mathcal{LFEFU}[(N, u_N)].$$

Observe that the second equivalence above is actually an isomorphism, as

shown by Proposition 3.5.

## Conclusions

In this paper we have shown how the unfolding semantics given in [12, 13] can be reconciled with a process-oriented semantics. Moreover, we have seen that the algebraic structure of the processes of a net can be faithfully expressed by a symmetric monoidal category. The key of these formal achievements is the notion of *decorated occurrence net*. Although DecOcc arose from the need of factorizing the involved adjunction from PTNets to Occ, and, thus, decorated occurrence nets might at first seem to be just a convenient technical solution, we have shown that in fact useful insights on the semantics of nets are suggested by the present unfolding construction and the associated notion of decorated occurrence net. In fact, decorated *deterministic* occurrence nets, suitably axiomatized as arrows of the symmetric monoidal category  $\mathcal{DP}[N]$ , provide both the process-oriented and the algebraic counterpart of the unfolding semantics. Moreover, they can be characterized as the minimal refinement of Goltz-Reisig processes which guarantees the identity of all tokens, i.e., as the minimal refinement of occurrence nets which guarantees the existence of an unfolding for PT nets. In fact, in order to achieve this it is necessary to disambiguate both the tokens in the same place of the initial marking and the tokens which are multiple instances of the same place, and, therefore, to introduce the notion of *families*.

## References

- [1] G. BERRY. Stable Models of Typed  $\lambda$ -calculi. In *Proceedings ICALP '78*, LNCS, n. 62, pp. 72–89, 1978.
- [2] E. BEST AND R. DEVILLERS. Sequential and Concurrent Behaviour in Petri Net Theory. *Theoretical Computer Science*, n. 55, pp. 87–136, 1987.
- [3] C. BROWN AND D. GURR. A Categorical Linear Framework for Petri Nets. In *Proceedings of the 5th LICS Symposium*, pp. 208–218, 1990.
- [4] C. BROWN, D. GURR, AND V. DE PAIVA. *A Linear Specification Language for Petri Nets*. Technical Report DAIMI PB-363, Computer Science Department, Aarhus University, 1991.
- [5] P. DEGANO, R. DE NICOLA, AND U. MONTANARI. A Distributed Operational Semantics for CCS based on Condition/Event Systems. *Acta Informatica*, n. 26, pp. 59–91, 1988.
- [6] P. DEGANO, J. MESEGUER, AND U. MONTANARI. Axiomatizing Net Computations and Processes. In *Proceedings of the 4th LICS Symposium*, pp. 175–185, IEEE, 1989.
- [7] J. ENGELFRIET. Branching Processes of Petri Nets. *Acta Informatica*, n. 28, pp. 575–591, 1991.
- [8] U. GOLTZ AND W. REISIG. The Non-Sequential Behaviour of Petri Nets. *Information and Computation*, n. 57, pp. 125–147, 1983.
- [9] P.W. HOOGERS, H.C.M. KLEIJN, AND P.S. THIAGARAJAN. Local Event Structures and Petri Nets. In *Proceedings of CONCUR '93*, LNCS, n. 715, pp. 462–476, Springer-Verlag, 1993.
- [10] S. MACLANE. *Categories for the Working Mathematician*. Springer-Verlag, 1971.
- [11] J. MESEGUER AND U. MONTANARI. Petri Nets are Monoids. *Information and Computation*, n. 88, pp. 105–154, Academic Press, 1990.
- [12] J. MESEGUER, U. MONTANARI, AND V. SASSONE. On the Semantics of Petri Nets. In *Proceedings of CONCUR '92*, LNCS, n. 630, pp. 286–301, Springer-Verlag, 1992.
- [13] J. MESEGUER, U. MONTANARI, AND V. SASSONE. *On the Semantics of Place/Transition Petri Nets*. Technical Report TR 27/92 Dipartimento Informatica, Università di Pisa, 1992, and Technical Report SRI-CSL-92-09, SRI International, Computer Science Laboratory, 1992.
- [14] J. MESEGUER, U. MONTANARI, AND V. SASSONE. *Process versus Unfolding Semantics for Place/Transition Petri Nets*. To appear as Technical Report of the Dipartimento di Informatica, Università di Pisa, and Technical Report of the Computer Science Laboratory, SRI International, 1994.
- [15] M. NIELSEN, G. PLOTKIN, AND G. WINSKEL. Petri Nets, Event Structures and Domains, Part 1. *Theoretical Computer Science*, n. 13, pp. 85–108, 1981.

- [16] M. NIELSEN, G. ROZENBERG, AND P.S. THIAGARAJAN. Behavioural Notions for Elementary Net Systems. *Distributed Computing*, n. 4, pp. 45–57, 1990.
- [17] E.R. OLDEROG. A Petri Net Semantics for CCSP. In *Advances in Petri Nets 1986, part II*, LNCS, n. 255, pp. 196–223, Springer-Verlag, 1987.
- [18] C.A. PETRI. *Kommunikation mit Automaten*. PhD Thesis, Institut für Instrumentelle Mathematik, Bonn, Germany, 1962.
- [19] C.A. PETRI. Concepts of Net Theory. In proceedings of *MFCS '73*, pp. 137–146, Mathematics Institute of the Slovak Academy of Science, 1973.
- [20] C.A. PETRI. *Non-Sequential Processes*. Interner Bericht ISF-77-5, Gesellschaft für Mathematik und Datenverarbeitung, Bonn, FRG, 1977.
- [21] W. REISIG. *Petri Nets*. Springer-Verlag, 1985.
- [22] V. SASSONE. *On the Semantics of Petri Nets: Processes, Unfoldings, and Infinite Computations*. PhD Thesis, TD 6/94 Dipartimento di Informatica, Università di Pisa, March 1994.
- [23] V. SASSONE. *Some Remarks on Concatenable Processes*. Technical Report TR 6/94, Dipartimento di Informatica, Università di Pisa, April 1994.
- [24] V. SASSONE, J. MESEGUER, AND U. MONTANARI.  $\omega$ -Ind Completion of Monoidal Categories and Infinite Petri Net Computations. Presentation at the *Workshop on Topology and Completion in Semantics*, Institute Blaise Pascal, November 1993, Chartres, France. To appear as Technical Report of the Dipartimento di Informatica, Università di Pisa, and Technical Report of the Computer Science Laboratory, SRI International, 1994.
- [25] D. SCOTT. Outline of a Mathematical Theory of Computation. In *Proceedings of 4th Annual Princeton Conference on Information Science and Systems*, pp. 169–176, 1970.
- [26] R. VAN GLABBEK AND F. VAANDRAGER. Petri Net Models for Algebraic Theories of Concurrency. In *proceedings of PARLE*, LNCS, n. 259, pp. 224–242, Springer-Verlag, 1987.
- [27] G. WINSKEL. Event Structure Semantics of CCS and Related Languages. In *Proceedings of ICALP '82*, LNCS, n. 140, pp. 561–567, Springer-Verlag, 1982. Expanded version available as technical report DAIMI PB-159, Computer Science Department, Aarhus University.
- [28] G. WINSKEL. Event Structures. In proceedings of *Advances in Petri Nets 1986, part II*, LNCS, n. 255, pp. 325–392, Springer-Verlag, 1987.
- [29] G. WINSKEL. Petri Nets, Algebras, Morphisms and Compositionality. *Information and Computation*, n. 72, pp. 197–238, 1987.