A class of hypernets with token creation and decidable reachability problem

Marek A Bednarczyk[1],[2], Piotr Jóźwiak[3], and Wiesław Pawłowski[1],[3]

[1] IPI PAN, Gdańsk, Poland
[2] ZWI PJWSTK, Gdańsk, Poland
[3] II UG, Gdańsk, Poland

Abstract. The formalism of Petri hypernets is extended by allowing creation of unstructured tokens. Any two such tokens are then declared indistinguishable, unless they have different sorts, i.e., *colours*. Under this assumption it is possible to provide a construction that to each coloured Petri hypernet associates a P/T net with an isomorphic case graph. Hence, the reachability problem for coloured Petri hypernets is decidable.

1 Introduction

Let us consider a high level Petri net formalism in which the nets manipulate tokens which are references to some *objects*. Following [8] let us assume that the following conditions are satisfied.

1. Within the formalism the "references to objects can be checked for identity".
2. The net should create *fresh* objects whenever creation takes place.

Kummer shows that under these assumptions the problem of reachability of the *empty marking* reduces to the problem of reachability of 2-counter automata, and thus it is undecidable.

In formalism introduced in [8] the first condition is implemented as follows. A variable $v_{x,y}$ is associated to every arrow between nodes x and y in the net. Then, a transition t is enabled in a marking M only when there is a map ζ which associates objects to variables so that from each input place p of t the map choses object $\zeta(v_{s,t})$ in M. Suppose then that the same variable labels arrows from two or more input places of t. Then the transition may fire only if there are references to the same object in all these places. Thus, such object nets operate on the simple type of abstract references with equality.

It turns out that undecidability of reachability problem holds for many non-trivial variants of higher order nets, and for nets-within-nets approaches in particular. This includes many classes of object nets studied in Hamburg, and nested Petri nets introduiced by Lomazova, see [10, 11].

On the other hand Petri hypernets, see [2], do not admit such testing for equality. An instructive way to compare the formalism proposed by Kummer with hypernets is to say that in hypernets each transition has an injective labeling of input and output arrows. In hypernets this is done by representing an object net as a synchronous product of state machine nets. Consequently, one cannot use such equality tests as described above. This gives hope that an extension which allows creation might not yield a framework with undecidable reachability problem. We introduce *coloured Petri hypernets*, abbr.: cPH, as an extension of Petri hypernets. This is done by admiting private transitions in agents which, when fire, create unstructured tokens.

It is assumed that tokens created in this way are distinguishable only if they have different sorts—each agent in a hypernet has a sort assigned to it. Thus, the sorts play the rôle of colours. This seems a right thing to do in order to maintain finitary character of the formalism. Besides, in [8] the reduction goes via reference to the reachability of the empty marking.

In an earlier paper [1] it has been shown that each marked hypernet can be translated to a 1-safe net system with an isomorphic case graph. Here we extend this construction and define a translation that maps a marked cPH to a P/T net system in such a way that their state spaces are isomorphic. This reduces the reachability problem for coloured Petri hypernets to that of P/T nets. Whence the reachability problem for coloured Petri hypernets is also decidable.

In what follows standard notational conventions regarding Petri nets and elementary set theory are used. For instance, $|A|$ denotes the cardinality of a set A, a singleton set is identified with its content.

The paper is organised as follows. Section 2 recalls the formalism of Petri hypernets. Section 3 introduces the formalism of coloured Petri hypernets. Section 4 provides the translation and presents the main result of the paper.

2 Petri hypernets

Our idea is to modify the definition of Petri hypernets given in [1] as little as possible so that creation of agents is allowed. First, we explain the static structure, and then the dynamic behaviour of hypernets.

2.1 The static structure of Petri hypernets

Let us start by recalling some terminology and definitions from [1]. A hypernet is a finite set of *agents*. A (mobile) agent is a synchronous product of a finite set of *modules*. A module is a state-machine net. Thus, in an agent each transition has the same number of pre- and post-conditions.

The state of a hypernet, called *hypermarking*, is given as a distribution of agents among places (also called *locations*) of other agents. Thus, in comparison with the classical nets-within-nets approaches, [13, 7], the only novelty so far is the requirement that our agents are explicitly decomposed into synchronous products of their *sequential components*. Note that by [5] any elementary net admits such a decomposition, and the same holds for 1-safe Petri nets, see [3].

Another restriction imposed on hypernets is that each agent and each module has assigned a *sort*. Let Σ be a finite set of sorts, and let α, β, ... range over Σ. We require that an agent of a sort α can be hosted only by the places of modules of sort α.

An extension of the classical idea comes with the notion of *communication ports*. Formally, let $I = \{?\} \cup \{?\beta \,|\, \beta \in \Sigma\}$ and $O = \{!\} \cup \{!\beta \,|\, \beta \in \Sigma\}$ be the sets of *input* and *output ports*, respectively. The purpose of ports is to allow for the exchange of tokens, i.e. agents, between agents. This allows the hierarchy of agents to evolve, unlike in some other approaches based on nets-within-nets paradigm.

Definition 1. *A* module *of sort α (α-module) is a, possibly empty, finite Petri net $N = (P, T, F)$, where P is the set of* local places*, T is the set of* transitions*, and $F \subseteq ((P \cup I) \times T) \cup (T \times (P \cup O))$ is the flow relation. We assume that $P \cap (I \cup O) = \emptyset$ and $|\mathsf{pre}\,(t)| = 1 = |\mathsf{post}\,(t)|$ for all $t \in T$.*

The last condition above says that each transition has exactly one pre- and one post-condition, either a local place, or a communication port. This condition is relaxed in the sequel to allow creation of agents.

Definition 2. *An* agent *$A = \{N^\alpha_A \,|\, \alpha \in \Sigma\}$ is a Σ-indexed set of modules, We call N^α_A the α-module of A, and assume $P^\alpha_A \cap P^\kappa_A = \emptyset$ whenever $\alpha \neq \kappa$.*

Modules should satisfy the following structural consistency conditions.

$$F^\alpha_A(?\beta, t) \quad \textit{implies} \quad \alpha \neq \beta \wedge F^\beta_A(p, t) \textit{ for some } p \in P^\beta_A, \tag{1}$$

$$F^\alpha_A(t, !\beta) \quad \textit{implies} \quad \alpha \neq \beta \wedge F^\beta_A(p, t) \wedge F^\beta_A(t, q) \textit{ for some } p, q \in P^\beta_A. \tag{2}$$

Above P^α_A, T^α_A and F^α_A, denote the respective components of an α-module of A.

In the sequel we let $P_A \mathrel{\widehat{=}} \bigcup\{P^\alpha_A \,|\, \alpha \in \Sigma\}$ and $T_A \mathrel{\widehat{=}} \bigcup\{T^\alpha_A \,|\, \alpha \in \Sigma\}$ denote the sets of local places and transitions of A, respectively.

Different modules do not share locations, but they can share transitions. Shared transition are used for the synchronisation of the respective modules. When each module of an agent is the empty net, then the agent behaves like an unstructured token. This observation will serve as the basis for identification of such unstructured tokens later on.

Ports ? and ! serve to communicate with agent's current *super-agent*. Ports of the form $?\beta$ and $!\beta$ serve to communicate with agent's *sub-agents* located in its β-module. In the sequel the two kinds of ports serve rôles complementary to each other. Let us consider first moving a token agent up in the hierarchy. Then, for instance, an agent can perform action t with $F^\alpha_A(?\beta, t)$ only together with one of its sub-agents in its β-module, and provided the sub-agent can perform t with $F^\alpha_A(t, !)$.

Conversly, an agent of sort β can perform t with $F_A^\alpha(t, !)$ only together with its super-agent, and provided its super-agent can perform t with $F_A^\alpha(?\beta, t)$. Similar cooperation must take place when moving tokens down the hierarchy, e.g., between an agent which wants to perform action t with $F_A^\alpha(t, !\beta)$ and its sub-agent in its β-module which intends to perform t with $F_A^\alpha(?, t)$. This idea is formally captured in the sequel by the notion of *consortium*.

To cope with graphical representation of ports the classical notation is somewhat extended. Consider the meaning of Def. 2(1) visualised on Fig. 1. It deals with communication of an agent with its tokens, i.e., sub-agents. Assume that α-module of an agent contains transition t with flow $F_A^\alpha(?\beta, t)$, i.e., t intends to get an α-agent from within its sub-agent stored in β-module. Then Def. 2(1) asserts that β must be different from α, and that the β-module must contain transition t, and the precondition of t in the β-module must not be a port.

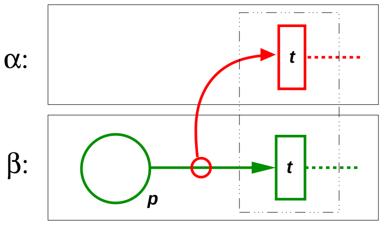

Fig. 1. Structural consistency between modules of an agent.

Thus, the consistency of a request by α-module to receive something from within agents in the β-module can be locally verified by looking at the agents in the local place p such that $F_A^\beta(p, t)$. Condition Def. 2(2) states that when agent wants to send something down, then t must be present in the β-module. Moreover, pre- and post-condition of t in β module must be locations, not ports.

We use a convention to draw $F_A^\alpha(?\beta, t)$ and $F_A^\alpha(t, !\beta)$ as links between the occurrence of t in the α-module and the input and output arcs of t in the β-module. Intended communication with super-agents is graphically represented by broken line half-circles, see Fig. 2.

Given agent A and any $t \in T_A^\alpha$ let $\mathsf{pre}_A^\alpha(t)$ and $\mathsf{post}_A^\alpha(t)$ denote the (necessarily unique) pre- and post-condition of t in the α-module of A, respectively. Note that $\mathsf{pre}_A^\alpha(t)$ and $\mathsf{post}_A^\alpha(t)$ can be either local places or communication ports.

Airport example, see [1] *Suppose one wants to model an airport as an agent in which two types of agents are to be handled:* travellers *and* planes*. A suitable hypernet should thus contain at least three sorts: let π be the sort for planes, τ for passengers and α for airports.*

Fig. 2 describes a possible structure of an airport agent which consists of two non empty modules: π-module for plane handling, and τ-module for travelers.

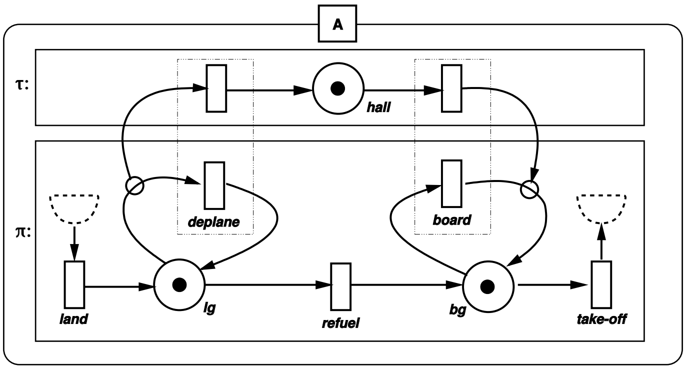

Fig. 2. Structure of an airport agent.

The π-module captures the usual plane handling procedure: plane lands from above; disposes of its passengers (if any); refuels; boards new passengers and finally takes off. Some of these activities are captured by transitions like land *and* $\mathsf{take\text{-}off}$ *which use communication with the implicit super-agent of the airport agent to move plane agents into and out of the airport. Transition* refuel*, which moves plane agents between locations of the airport agent does not require* any *synchronisation. Finally, transitions* $\mathsf{deplane}$ *and* board *are present in both modules and their firing does require synchronisation. Intuitively, to fire* board*, say, we need a traveler agent in location* hall *and a plane agent in location* bg*. Moreover, the transition in the τ-module has an output port as the postcondition, so the plane agent should be prepared to engage in* board *with flow $F(?, \mathsf{board})$.*

Definition 3. *A* Petri hypernet *is a finite family $\mathcal{N}$ of agents together with* sorting*, i.e., a function $\sigma : \mathcal{N} \to \Sigma$ that assigns to each agent its sort.*

It is assumed that the agents in $\mathcal{N}$ have mutually disjoint sets of local places. Thus, any place $p \in P_{\mathcal{N}} \mathrel{\widehat{=}} \bigcup\{P_A \mid A \in \mathcal{N}\}$ can be assigned its sort $\sigma(p) = \alpha$ where α is the sort of a module of the (unique) $A \in \mathcal{N}$, such that $p \in P^{\alpha}_{A}$.

In hypernets the locations of an agent are distributed among its modules. Hence, the state of a hypernet is uniquely defined as a *many-to-one* correspondence between agents and locations, cf. [2].

Definition 4. *The state of a hypernet, called* hypermarking*, is a sort-preserving partial function $\mu : \mathcal{N} \rightharpoonup P_{\mathcal{N}}$.*

2.2 The dynamics of Petri hypernets

The dynamics of hypernet reflects their ability to change *state* of the system.

Suppose $t \in T_A$ is a transition of agent A. Then A in a given state can perform t only if all its α-modules such that $t \in T^{\alpha}_{A}$ can perform t. This means that we assume inter-module synchronisation on common transitions.

Let us concentrate now on a single α-module of A such that $t \in T^{\alpha}_{A}$. If the input and output places of t in the module are locations, i.e., local places, then in order to perform t it is enough to find a token in $\mathsf{pre}^{\alpha}_{A}(t)$ and move it to $\mathsf{post}^{\alpha}_{A}(t)$. Otherwise, either $\mathsf{pre}^{\alpha}_{A}(t)$ or $\mathsf{post}^{\alpha}_{A}(t)$, or both, are communication ports. Then inter-level synchronisation between agents should take place.

Flow of tokens and consortia.

The inter-module synchronisation is *static* in the sense that it concernes a fixed set of modules of an agent. The inter-agent interaction is *dynamic*—it depends on the current hypermarking—distribution of agents among locations. Our aim is to explain how such a distribution may evolve.

Consider agents $\mathbf{A}$ and $\mathbf{A}'$ which try to help each other to perform transition t by establishing a communication link for α-tokens flow from $\mathbf{A}$ to $\mathbf{A}'$, notation $\mathbf{A} \rhd_{\alpha} \mathbf{A}'$. Such flow can go either up or down the hierarchy. The first case is described on the left, the second on the right of Fig. 3. The existence of an α-flow requires that two conditions hold.

Adjacency: One of the agents should be the super-agent of the other.
Matching: Each agent should match the communication request of the other.

Airport example (cntd.) *In Fig. 4 the airport agent contains two plane agents* $\mathbf{P1}$ *and* $\mathbf{P2}$*, and also two traveller agents* $\mathbf{T1}$ *and* $\mathbf{T2}$*. It should be clear that* $\mathbf{P1} \rhd_{\tau} \mathbf{A}$ *and* $\mathbf{A} \rhd_{\tau} \mathbf{P2}$ *hold.*

Fig. 4 also demonstrates that in hypernets inter-module and inter-agent synchronisation may take place at the same time. Thus, one is interested not just in pairs of agents, but in groups of agents which should cooperate with each other to ensure that together *all* their synchronisation and communication requests related to execution of a shared transition are satisfied.

Let $\mathcal{N}$ be a hypernet with sorting σ and consider a triple $\Gamma = (t, \mathcal{T}, \xi)$ which satisfies the following.

- $t \in T_{\mathcal{N}} \mathrel{\widehat{=}} \bigcup\{T_A \mid A \in \mathcal{N}\}$ is a transition of $\mathcal{N}$.
- $\mathcal{T} \subseteq \mathcal{N}$ is the set of *active* agents of Γ.
  All active agents share t, i.e., $\mathcal{T} \subseteq \{A \in \mathcal{N} \mid t \in T_A\}$.

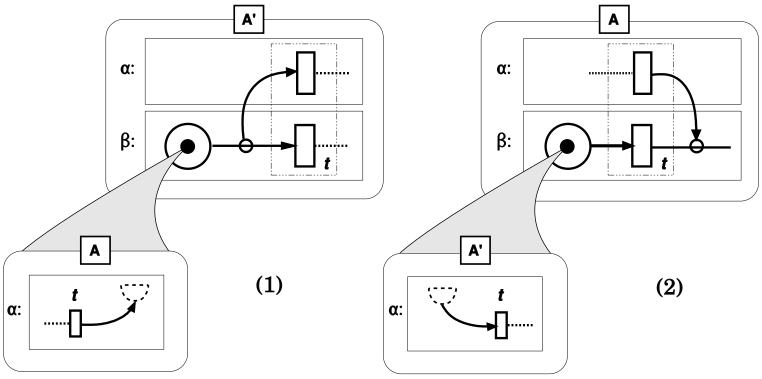

Fig. 3. Two instances of inter-level communication flow $\mathbf{A} \rhd_\alpha \mathbf{A}'$ from $\mathbf{A}$ to $\mathbf{A}'$. Flow up on the left, flow down on the right

- $\mathsf{In}(\Gamma) \mathrel{\widehat{=}} P_{\mathcal{N}} \cap \bigcup_{\alpha\in\Sigma} \{\mathsf{pre}^\alpha_A(t) \,|\, A \in \mathcal{T}\}$ is the set of *local inputs* of Γ.
- $\mathsf{Out}(\Gamma) \mathrel{\widehat{=}} P_{\mathcal{N}} \cap \bigcup_{\alpha\in\Sigma} \{\mathsf{post}^\alpha_A(t) \,|\, A \in \mathcal{T}\}$ is the set of *local outputs* of Γ.
- $\xi : \mathsf{In}(\Gamma) \to \mathcal{N}$ is *token choice* function of Γ.
  ξ is injective and sort preserving: $\sigma(\xi(p)) = \sigma(p)$ for each place $p \in \mathsf{In}(\Gamma)$.
- $\mathcal{T}_\xi = \{\xi(p) \,|\, p \in \mathsf{In}(\Gamma)\}$ is the set of *passive agents* or *tokens* of Γ.
  Note that an agent can be active and passive at the same time.
- $\uparrow : \mathcal{T}_\xi \to \mathcal{T}$ yields super-agents for each token of Γ.
  Formally, $A' \uparrow = A$ iff $\xi(p) = A'$ for some $p \in P_A$.

Definition 5. *Consider Γ as above and active agents $A, A' \in \mathcal{T}$. Then, there is an α-*flow *from A to A' in Γ, written $A \rhd_\alpha A'$, if one of the following conditions is satisfied.*

flow up $\mathsf{post}^\alpha_A(t) = !$ *and* $\xi(\mathsf{pre}^\beta_{A'}(t)) = A$ *and* $\mathsf{pre}^\alpha_{A'}(t) = ?\beta$ *and* $\beta = \sigma(A)$.
flow down $\mathsf{post}^\alpha_A(t) = !\beta$ *and* $\xi(\mathsf{pre}^\beta_A(t)) = A'$ *and* $\mathsf{pre}^\alpha_{A'}(t) = ?$ *and* $\beta = \sigma(A')$.

Finally, we are ready to define the notion of consortium.

Definition 6. *A triple Γ as above is a* consortium *provided the following hold.*

1. *$\mathcal{T}$ is non-empty.*
2. *$A \in \mathcal{T}$ and $F^\alpha_A(?, t)$ implies $A \in \mathcal{T}_\xi$ and $A\!\uparrow \rhd_\alpha A$*
3. *$A \in \mathcal{T}$ and $F^\alpha_A(t, !)$ implies $A \in \mathcal{T}_\xi$ and $A \rhd_\alpha A\!\uparrow$*
4. *$A \in \mathcal{T}$ and $F^\alpha_A(?\beta, t)$ implies $\xi(p) \in \mathcal{T}$ and $\xi(p) \rhd_\alpha A$, where $p = \mathsf{pre}^\beta_A(t)$*
5. *$A \in \mathcal{T}$ and $F^\alpha_A(t, !\beta)$ implies $\xi(p) \in \mathcal{T}$ and $A \rhd_\alpha \xi(p)$, where $p = \mathsf{pre}^\beta_A(t)$*
6. *The undirected graph $(\mathcal{T}, \bigcup_{\alpha\in\Sigma} \rhd_\alpha)$ is connected.*

Most of the conditions have a simple justification. Condition 1 says that at least one active agent has to be involved. Conditions 2-5 stipulate that members of $\mathcal{T}$ provide matching communication capabilities for other members of the group. Note that structural consistency conditions of Def. 2 guarantee the existence of a suitable location p. Finally, condition 6 says that $\mathcal{T}$ is a minimal such group of agents.

Note that transaction may involve just one active agent, see for instance transition *refuel* in our running example.

Airport example (cntd.). *Let us consider airport agent from Fig. 2 together with two airplane agents* **P1** *and* **P2***, and two traveller agents* **T1** *and* **T2***. The plane agents have a very simple structure. It consists of just one non empty module for traveller handling. A plane is always ready to accept a new passenger from its super-agent, and always ready to dispose of any travellers onboard. Traveller agents are unstructured tokens.*

The group together with two consortia is depicted on Fig. 4. One consortium involves deplane *and agents* **A** *and* **P1** *as active agents. The local input places of these agents are* lg *and* c1*. The*

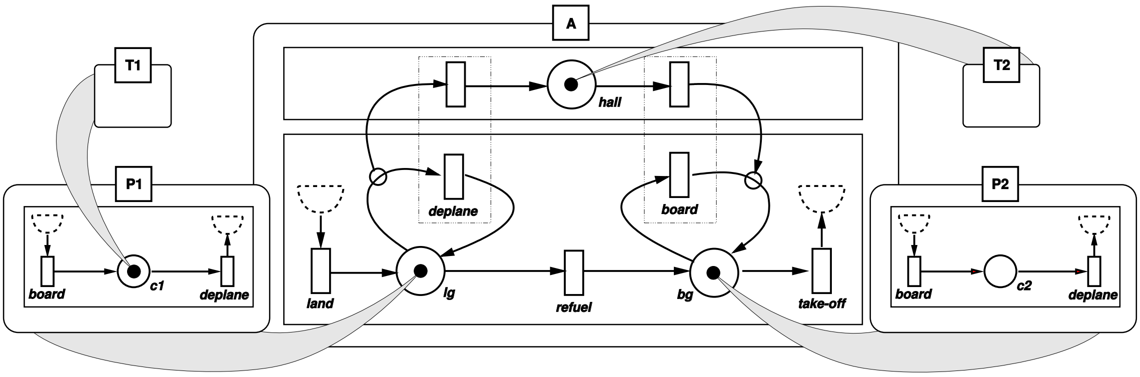

Fig. 4. Inter-level agent communication flow: $\mathbf{P1} \rhd_\tau \mathbf{A}$ and $\mathbf{A} \rhd_\tau \mathbf{P2}$.

passive token agents chosen for these places are **P1** *and* **T1***, respectively. Note that* **P1** *is both active and passive. This means that it will move a token, here* **T1***, and be moved as a token at the same time. There is one flow involved in this consortium, namely* $\mathbf{P1} \rhd_\tau \mathbf{A}$*.*

Another consortium involves board *and agents* **A** *and* **P2** *as active agents. The local input places of these agents are* bg *and* l*. The passive token agents chosen for these places are* **P2** *and* **T2***, respectively. The only flow involved is* $\mathbf{A} \rhd_\tau \mathbf{P2}$*.*

Chains of flow.

A consortium $\mathit{\Gamma} = (t, \mathcal{T}, \xi)$ is a minimal group of vertically interconnected agents which together can change the state of the hypernet by moving some agents from one location to another. To formalise this idea we need to know *which* agents to move, and also their *initial* and *final* locations.

Intuitively, agents in $\mathcal{T}$ move those in $\mathcal{T}_\xi$. The initial locations of token agents $\mathcal{T}_\xi$ is given by ξ^{-1}. So, the only problem is to find their final destination. The ability to find such unique locations stems from conditions 2-5 of Definition 6. The mechanism involved is explained by means of a generalisation of the notion of α-flow among the members of consortium $\mathit{\Gamma}$.

Definition 7. *An α-*chain of flow *in consortium $\mathit{\Gamma}$ is a finite non-empty sequence $c = A_1 \ldots A_n$ of agents from $\mathcal{T}$, such that $A_i \rhd_\alpha A_{i+1}$ for $1 \leq i < n$.*

Let $\mathsf{pre}^\alpha(c) \mathrel{\widehat{=}} \mathsf{pre}^\alpha_{A_1}(t)$ and $\mathsf{post}^\alpha(c) \mathrel{\widehat{=}} \mathsf{post}^\alpha_{A_n}(t)$.

Proposition 1. *Let $\mathit{\Gamma} = (t, \mathcal{T}, \xi)$ be a consortium. For any agent $A \in \mathcal{T}$ such that $\mathsf{pre}^\alpha_A(t)$ is a* location *there exists a unique α-chain $c = A_1 \ldots A_n$ such that $A = A_1$ and $\mathsf{post}^\alpha_{A_n}(t)$ is a local place as well.*

Given an output place of $\mathit{\Gamma}$ one can start building a chain of flow which will lead to a corresponding input location.

Proposition 2. *Each consortium $\mathit{\Gamma} = (t, \mathcal{T}, \xi)$ induces a sort-preserving bijection $\mathbf{trg}_\mathit{\Gamma} : \mathsf{In}(\mathit{\Gamma}) \to \mathsf{Out}(\mathit{\Gamma})$ where $\mathbf{trg}_\mathit{\Gamma}(p) = q$ iff there is a κ-chain of flow c in $\mathit{\Gamma}$, where $\kappa = \sigma(p)$, such that $p = \mathsf{pre}^\kappa(c)$ and $q = \mathsf{post}^\kappa(c)$. Moreover, for $p \in \mathsf{In}(\mathit{\Gamma})$ if $\mathbf{trg}_\mathit{\Gamma}(p) \in \mathsf{In}(\mathit{\Gamma})$ then $\mathbf{trg}_\mathit{\Gamma}(p) = p$.*

Proof. The first part follows immediately by Proposition 1. So, assume $q = \mathbf{trg}_\mathit{\Gamma}(p) \in \mathsf{In}_\mathit{\Gamma}()$ for some $p \in \mathsf{In}(\mathit{\Gamma})$. Then, by definition of $\mathbf{trg}_\mathit{\Gamma}$ it follows that $q = \mathsf{post}^\alpha_A(t)$ for the last agent $A \in \mathcal{T}$ in any α-chain of flow that starts with p.. But from $q = \mathbf{trg}_\mathit{\Gamma}(p) \in \mathsf{In}(\mathit{\Gamma})$ it follows that $q = \mathsf{pre}^\alpha_A(t)$. So, this chain contains just A, and so $p = q$ follows. □

A consortium is enabled at a hypermarking when the choice of the passive tokens in consortium agrees with their location described by the hypermarking. The effect of firing enabled transition is to update the hypermarking by synchronously moving the passive agents to locations prescribed by **trg** function.

Definition 8. *Let $\mu : \mathcal{N} \rightharpoonup P_{\mathcal{N}}$ be a hypermarking of $\mathcal{N}$. Then consortium $\Gamma = (t, \mathcal{T}, \xi)$ is* enabled *at μ, notation $\mu[\Gamma\rangle$, if $\mu(\xi(p)) = p$ for all $p \in \mathsf{In}(\Gamma)$. Γ enabled at μ can* fire*, yielding a new hypermarking μ' defined as follows.*

$$\mu' A = \begin{cases} \mathbf{trg}_{\Gamma}(\mu A) & A \in \mathcal{T}_{\xi} \\ \mu A & A \notin \mathcal{T}_{\xi} \end{cases}$$

Consider a hypernet $(\mathcal{N}, \sigma)$ with an initial hypermarking μ. Then its *case graph* has hypermarkings reachable from μ as states and firings of consortia as transitions.

Boarding a passenger—an example of hierarchy change

The example below demonstrates the effect of an inter-level communication between agents. A situation at the airport as described on Fig. 4, offers two possible consortia—corresponding to the *deplane* and *board* actions respectively. The latter consists of a plane **P2** and the airport **A** as active agents and a traveller **T2** and plane **P2** as passive tokens. As a result of firing the consortium the passenger **T2** is moved from the passenger's lounge (place **hall**) to the plane and vanishes from the level of the airport. Plane **P2** stays in place **bg**. The new situation is depicted on Fig. 5.

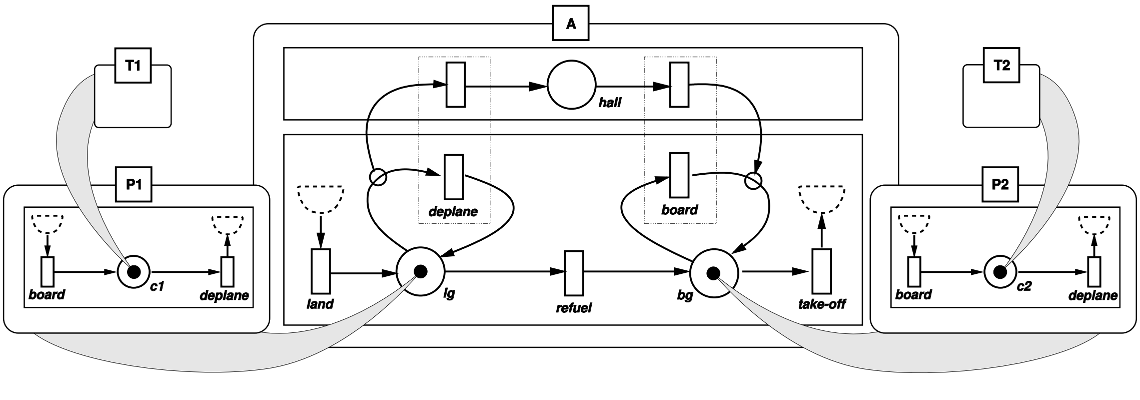

Fig. 5. T2 is boarding plane **P2**

Note that such a change is not supported by most theories that deal with nets-within-nets formalisms. On the other hand hierarchy change can be modelled in Renew, see [9].

3 Petri hypernets with creation of coloured tokens

Extension of a formalism in which individual agents may contain other agent rises basic yet important questions. Let us start by reviewing some of them.

- *Can the agents created be complex?* Only unstructured/empty agents can be created.
- *Do we allow destruction of agents?* Destruction is not allowed. Thus, we do not have to investigate what happens to an agent which is destroyed or to it's subagents.
- *How to cope with unbounded name spaces when creation is allowed?* Newly created agents of the same kind (sort) are indistinguishable.

Additionally, to simplify matters it is assumed that creation is local to an agent, i.e., agent interaction does not involve creation.

3.1 Static structure of coloured hypernets

Let us start by relaxing the Def. 1 by allowing transitions in a module to have *at most* one precondition and exactly one post-condition (to allow creation of tokens). The notion of agent given in Def. 2 remains unchanged. The definition of hypernet is modified as follows.

Definition 9. *A* coloured Petri hypernet *is a finite family $\mathcal{N}$ of agents together with* sorting, *i.e., a function $\sigma : \mathcal{N} \to \Sigma$ that assigns to each agent its sort. Given α, for any $t \in T_{\mathcal{N}}$ if $\mathsf{pre}^{\alpha}_{A}(t) = \emptyset$ for some $A \in \mathcal{N}$ then $t \in T_{A'}$ with $A' \in \mathcal{N}$ implies $A = A'$. Moreover, for any $\beta \in \Sigma$ the post-condition $\mathsf{post}^{\beta}_{A}(t)$ is a location, and and if $\mathsf{pre}^{\beta}_{A}(t)$ exists then $\mathsf{pre}^{\beta}_{A}(t)$ is a location as well.*

Thus, the restriction imposed on hypernets is that if a transition of an agent lacks a pre-condition in one of its modules then this is a private transition of that agent. Moreover, such transition has only locations as pre- and post-conditions. This assumption has important consequences. Now every consortium related to agent creating transition consists of a singleton active agent.

3.2 Coloured tokens and hypermarkings

In hypernets every agent has a sort assigned to it. We have postulated that agents which are created are unstructured, so let us refer to any such agent by means of its sort. This also allows to capture the postulate that created agents are indistinguishable: three such agents of sort σ will be represented as $3 \cdot \sigma$.

As a consequence of the above decision one has to modify the notion of hypermarking to allow both: distinguishable and indistinguishable agents in locations.

Definition 10. *The state of a coloured hypernet, called* hypermarking*, consists of a sort-preserving partial function $\mu : \mathcal{N} \rightharpoonup P_{\mathcal{N}}$ and a function $\nu : P_{\mathcal{N}} \to \mathbb{N}$.*

Thus, the old notion is extended by another component that simply counts how many unstructured tokens are stored in each location.

3.3 Coloured tokens and consortia

Since created agents are unstructured they cannot play the rôle of active agents. Thus, the notion of consortium for coloured hypernets is almost identical to the basic notion introduced in Def. 6.

The only difference is that now we require that the token choice function ξ for the triple $\Gamma = (t, \mathcal{T}, \xi)$ is *partial.* The idea here is that if $\xi(p)$ is undefined for some $p \in \mathsf{In}(\Gamma)$ then implicitly an unstructured agent is chosen.

Note also, that the idea of communication flow between agents involves only structured agents. Hence, the definition of flow in Γ, Def. 5, and the definition of consortium, Def. 6, stays the same. The same remark applies to the notion of chain of flow, cf. Def. 7.

Whence, Prop. 1 holds without change, whereas Prop. 2 in case of coloured Petri hypernets needs minute modification.

Proposition 3. *Each consortium $\Gamma = (t, \mathcal{T}, \xi)$ induces a sort-preserving injection $\mathbf{trg}_{\Gamma} : \mathsf{In}(\Gamma) \to \mathsf{Out}(\Gamma)$ where $\mathbf{trg}_{\Gamma}(p) = q$ iff there is a κ-chain of flow c in Γ, where $\kappa = \sigma(p)$, such that $p = \mathsf{pre}^{\kappa}(c)$ and $q = \mathsf{post}^{\kappa}(c)$. Moreover, for $p \in \mathsf{In}(\Gamma)$ if $\mathbf{trg}_{\Gamma}(p) \in \mathsf{In}(\Gamma)$ then $\mathbf{trg}_{\Gamma}(p) = p$.*

Proof. The same arguments as in the proof of Prop. 2. □

The only difference between the above and Prop. 2 is that now we cannot claim that $\mathbf{trg}_{\Gamma}$ is a bijection, since in the presence of agent creating transitions the caridnality of $\mathsf{In}(\Gamma)$ could be smaller than that of $\mathsf{Out}(\Gamma)$. Thus, a simple modification allows to define the dynamics of coloured Petri hypernets as follows.

Definition 11. *Let (μ, ν) be a hypermarking of cPH $\mathcal{N}$ with $\mu : \mathcal{N} \rightharpoonup P_{\mathcal{N}}$ and $\nu : P_{\mathcal{N}} \to \mathbb{N}$. Consider consortium $\Gamma = (t, \mathcal{T}, \xi)$. Then Γ is* enabled *at (μ, ν), notation $(\mu, \nu)[\Gamma\rangle$, if for all $p \in \mathsf{In}(\Gamma)$ either $\mu(\xi(p)) = p$ when $\xi(p)$-defined, or $\nu(p) > 0$ when $\xi(p)$-undefined.*

Γ enabled at (μ, ν) can fire*, yielding a new hypermarking (μ', ν') defined as follows.*

$$\mu' A = \begin{cases} \mathbf{trg}_{\Gamma}(\mu A) & A \in \mathcal{T}_{\xi} \\ \mu A & A \notin \mathcal{T}_{\xi} \end{cases}$$

$$\nu' q = \begin{cases} \nu q + 1 & q \in \mathsf{Out}(\Gamma) \setminus \mathsf{In}(\Gamma) \setminus \mathbf{trg}_{\Gamma}(\{p \in \mathsf{In}(\Gamma) \mid \xi(p)\text{-}\mathit{defined}\}) \\ \nu q - 1 & q \in \mathsf{In}(\Gamma) \setminus \mathsf{Out}(\Gamma) \wedge \xi(q)\text{-}\mathit{undefined} \\ \nu q & \text{otherwise} \end{cases}$$

The correctness of the above definition stems from Prop. 2.

3.4 Coloured hypernets—an example

Fig. 6 shows a variant of Airport hypernet in which two travellers agents with empty structure have been replaced by tokens. The coloured hypernet contains also a transition nT capable of creating agents. The effect of firing nT twice in a row from the given state is shown on Fig. 7.

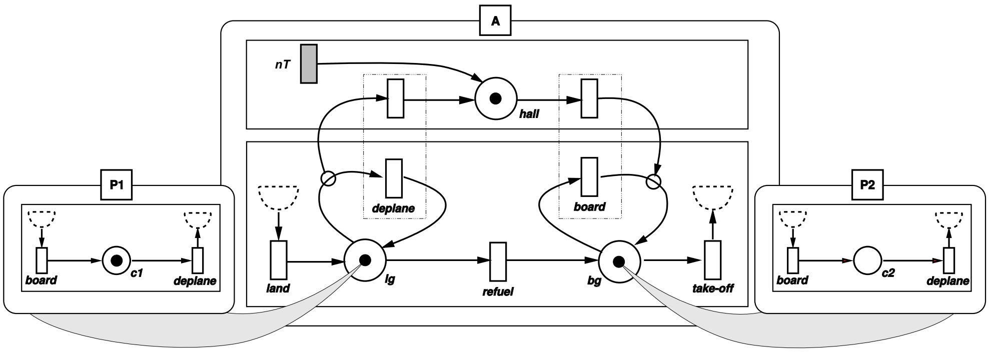

Fig. 6. A coloured version of Airport from Fig. 4

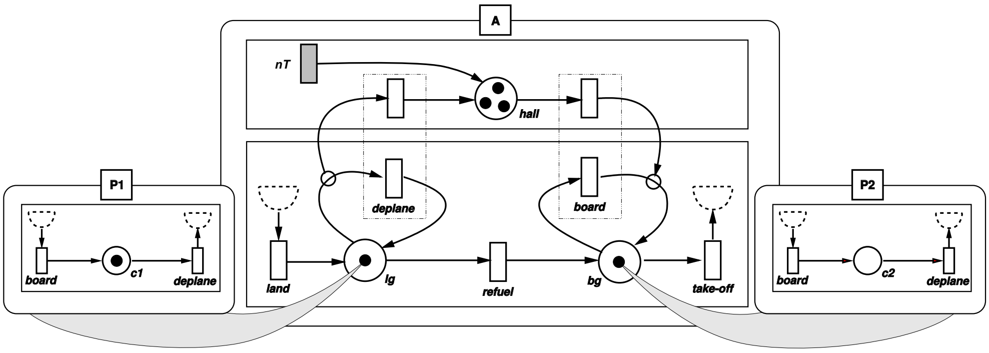

Fig. 7. Coloured token creation in Airport from Fig. 6

4 From coloured Petri hypernets to P/T nets

In this section we show that the semantics of coloured Petri hypernets can be explained by means of finite P/T nets, cf. [12]. The construction is done in two stages.

First, to each coloured hypernet $H = (\mathcal{N}, \sigma)$ we associate an unmarked Petri net N_H. Second, to each hypermarking (μ, ν) of H we associate a marking $M_{(\mu,\nu)}$ of N_H in such a way that the dynamic evolution of H from (μ, ν) is *isomorphic* to the dynamic evolution of N_H with $M_{(\mu,\nu)}$ as an initial marking.

4.1 Structural expansion of coloured Petri hypernets

Let $H = (\mathcal{N}, \sigma)$ be a coloured hypernet. Define $N_H = (P, T, F)$ as follows.

Places $P = C_H \cup B_H$ where
$C_H \mathrel{\widehat{=}} \{\langle p\rangle \mid p \in \bigcup\{P_A \mid A \in \mathcal{N}\}\}$,
$B_H \mathrel{\widehat{=}} \{\langle p, A\rangle \mid \sigma(p) = \sigma(A), A \in \mathcal{N}, p \in P_{\mathcal{N}}\}$.
We assume that $C_H \cap B_H = \emptyset$.
The intended meaning:

- Place $\langle p, A\rangle$ is used to mark presence of agent A at p in H.
- Place $\langle p\rangle$ is used to count the number of indistinguishable agents at p in H.

Transitions $T \mathrel{\widehat{=}} \{\Gamma \mid \Gamma \text{ is a consortium of } H\}$.

Flow relation $F \subseteq P \times T \cup T \times P$ defined as follows.
Let $\Gamma = (t, \mathcal{T}, \xi)$ be a consortium in H. Then
$F(\langle p, A\rangle, \Gamma)$ iff $p \in \mathsf{In}(\Gamma)$ and $\xi(p) = A$
$F(\Gamma, \langle p, A\rangle)$ iff $p \in \mathsf{Out}(\Gamma)$ and $\xi(\mathbf{trg}_{\Gamma}^{-1}(p)) = A$.
$F(\langle p\rangle, \Gamma)$ iff $p \in \mathsf{In}(\Gamma)$ and $\xi(p)$-undefined
$F(\Gamma, \langle p\rangle)$ iff $p \in \mathsf{Out}(\Gamma) \setminus \mathbf{trg}_{\Gamma}(\mathsf{In}(\Gamma))$ or $\xi(\mathbf{trg}_{\Gamma}^{-1}(p))$-undefined.

Thus, each location p of H is modeled in N_H by a number of copies. First, there is $\langle p, A\rangle$ for each agent A of an appropriate sort. Secondly, there is $\langle p\rangle$ to cope with the indistinguishable agents that can be stored in p.

Just as in [1] consortia become transitions of the resulting net. Each consortium contains detailed information about the rôle of its active and passive agents, including the rôle of passive indistinguishable agents. This, on the basis of Prop. 3 and Def. 11, is reflected in the definition of flow of N_H above.

The effect of structural expansion of coloured hypernet from Fig. 6 gives rise to the marking depicted on Fig. 8.

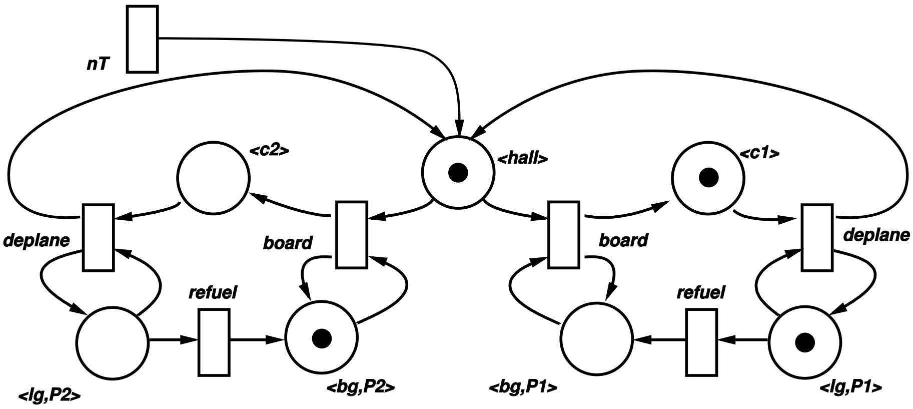

Fig. 8. Structural expansion of hypernet from Fig. 6

4.2 From marked coloured Petri hypernets to P/T net systems

Let (μ, ν) be a coloured hypermarking on H. Define marking $M_{(\mu,\nu)} : P \to \mathbb{N}$ of the corresponding Petri net P_H as follows.

$$M_{(\mu,\nu)}\langle p, A\rangle = \begin{cases} 1 & \mu A = p \\ 0 & \mu A \neq p \end{cases} \tag{3}$$

$$M_{(\mu,\nu)}\langle p\rangle = \nu p \tag{4}$$

It is not true that every marking on N_H can be obtained from a hypermarking by the above construction. For instance, markings of the form $M_{(\mu,\nu)}$ satisfy.

$$M\langle p, A\rangle \leq 1 \quad \text{and} \quad (\forall p, q \in P_{\mathcal{N}})\ M\langle p, A\rangle = 1 = M\langle q, A\rangle \Rightarrow p = q \tag{5}$$

If, however, a marking of N_H satisfies (5) then the converse construction can be given as follows.

$$\mu_M A = p, \text{ provided } M\langle p, A\rangle = 1 \tag{6}$$

$$\nu_M p = M\langle p\rangle \tag{7}$$

It should be evident that together the constructions define a bijection between hypermarkings of coloured Petri hypernet H and markings of N_H that satisfy (5). From this observation the main result follows easily.

Theorem 1. *The hyper-case graph of a marked coloured hypernet is isomorphic to the case graph of the corresponding Petri net system.*

Proof. Let H be a coloured Petri hypernet with hypermarking (μ, ν).

Then $(\mu, \nu)[\Gamma\rangle$ in H iff $M_{(\mu,\nu)}[\Gamma\rangle$ in N_H.

Moreover, $M[\Gamma\rangle M'$ in N_H and M satisfies (5) then M' satisfies (5) as well.

So the result follows since $(\mu, \nu)[\Gamma\rangle(\mu', \nu')$ in H implies $M_{(\mu,\nu)}[\Gamma\rangle M_{(\mu',\nu')}$ in N_H holds by construction for an arbitrary consortium Γ of H. □

On the basis of the above one can easily reduce the problem of reachability of a hypermarking from a given one to the problem of reachability in the world of P/T net systems. Since the latter is decidable, it follows that so is the former.

Corollary 1. *The reachability problem for coloured Petri hypernets is decidable.*

Clearly, the corollary has philospohical rather than practical impact since the complexity of the decidability procedure is very high. Additionally, the process of expansion could blow up the size of the hypernet.

The marked coloured hypernet from Fig. 7 gives rise to the marking depicted on Fig. 9 in the expanded version.

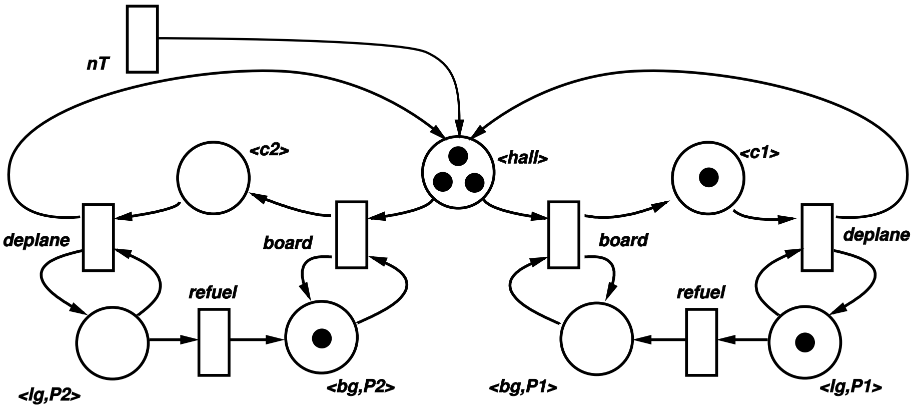

Fig. 9. Expanded marked hypernet from Fig. 7

5 Conclusions and future work

Petri hypernets were introduced inside the nets-within-nets paradigm in [2] as a visual framework suitable for modeling mobility of agents. The main difference with other approaches to mobility based on Petri nets is that in hypernets the hierarchy of agents may evolve.

The individual agent semantics of hypernets (each agent is identifiable) is another difference between hypernets and other models. The choice of this semantics leads to a logic in which the dynamic and structural properties of agents can be expressed and model checked as proposed in [4].

In [1] the authors explained Petri hypernets by means of a translation into 1-safe Petri nets. This translation not only shows that hypernets are well rooted inside the theory of Petri nets. Besides this foundational aspect, the translation allows us to apply the rich set of techniques and algorithms developed over the years for analyzing 1-safe nets (see, for example, [6, 12]). Among them, we find algorithms to check liveness, reachability of a place or of a marking, fairness properties, and so on. The family of methods based on the computation of place- and transition-invariants can then be applied in order, for instance, to check safety properties of a hypernet.

In this note an extension of Petri hypernets was introduced which allows creation of agents with empty structure. To bound the name space of the objects in the formalism we have imposed restriction that any two such agents of the same kind (sort) are not distinguishable.

With this assumption it was possible to provide a translation of the extended class of hypernets to P/T nets. Again, the translation establishes an isomorphism between states of a hypernet and a class of reachable markings of the corresponding P/T net. Thus, again, one can apply the theory developed to study P/T nets to cope with this class of hypernets.

Concerning future work—it seems that there should be no problem to allow consumption/destruction of agents together with creation. It seems, however, that this may be done in several ways. Here, the question to answer is: *What happens to sub-agents of an agent if the latter is destroyed?*

We plan also to investigate extensions of logic developed in [4] to hypernets with agent creation.

Formalisms based on the nets-within-nets paradigm often fall victims of undecidability of fundamental problems, like reachability. We plan to compare our approach with some of them, e.g., [7], to better understand the reasons.

References

1. Marek A. Bednarczyk, Luca Bernardinello, Wiesław Pawłowski, and Lucia Pomello. From Petri hypernets to 1-safe nets. In Daniel Moldt, editor, *Proceedings of the 4th International Workshop on Modelling of Objects, Components, and Agents, MOCA'06*, volume Bericht 272, FBI-HH-B-272/06, pages 23–43. Department für Informatik, Universität Hamburg, 2006.
2. Marek A. Bednarczyk, Luca Bernardinello, Wiesław Pawłowski, and Lucia Pomello. Modelling mobility with Petri hypernets. In José Luiz Fiadeiro, Peter D. Mosses, and Fernando Orejas, editors, *Recent Trends in Algebraic Development Techniques. 17th International Workshop WADT 2004, Barcelona, Spain, March 27-30, 2004. Revised Selected Papers*, volume 3423 of *LNCS*, pages 28–44. Springer-Verlag, 2005.
3. Marek A. Bednarczyk and Andrzej M. Borzyszkowski. On concurrent realization of reactive systems and their morphisms. In Hartmut Ehrig, Gabriel Juhás, Julia Padberg, and Grzegorz Rozenberg, editors, *Unifying Petri Nets*, volume 2128 of *Lecture Notes in Computer Science*, pages 346–379. Springer, 2001.
4. Marek A. Bednarczyk, Wojciech Jamroga, and Wiesław Pawłowski. Expressing and verifying temporal and structural properties of mobile agents. In Ludwik Czaja, editor, *Proceedings of the Concurrency, Specification and Programming Workshop CS&P'05*, pages 57–68, 2005.
5. Luca Bernardinello. Synthesis of net systems. In Marco Ajmone Marsan, editor, *Application and Theory of Petri Nets*, volume 691 of *Lecture Notes in Computer Science*, pages 89–105. Springer, 1993.
6. Javier Esparza. Decidability and complexity of Petri net problems - an introduction. In Reisig and Rozenberg [12], pages 374–428.
7. Michael Köhler and Heiko Rölke. Properties of object Petri nets. In Wil van der Aalst and Eike Best, editors, *Applications and Theory of Petri Nets 2004, Proceedings*, volume 3099 of *LNCS*, pages 278–297. Springer-Verlag, 2004.
8. Olaf Kummer. Undecidability in object-oriented Petri nets. *Petri Net Newsletter*, 59:18–23, 2000.
9. Olaf Kummer, Frank Wienberg, Michael Duvigneau, Jörn Schumacher, Michael Köhler, Daniel Moldt, Heiko Rölke, and Rüdiger Valk. An extensible editor and simulation engine for petri nets: Renew. In Jordi Cortadella and Wolfgang Reisig, editors, *ICATPN*, volume 3099 of *Lecture Notes in Computer Science*, pages 484–493. Springer, 2004.
10. Irina A. Lomazova. Nested petri nets — a formalism for specification of multi-agent distributed systems. *Fundamenta Informaticae*, 43((1-4)):195–214, 2000.
11. Irina A. Lomazova and Philippe Schnoebelen. Some decidability results for nested petri nets. *Lecture Notes in Computer Science*, 1755:208–220, 2000.
12. Wolfgang Reisig and Grzegorz Rozenberg, editors. *Lectures on Petri Nets I: Basic Models, Advances in Petri Nets*, volume 1491 of *LNCS*. Springer, 1998.
13. Rüdiger Valk. On processes of object Petri nets. Technical Report FB-185, Fachbereich Informatik, Universität Hamburg, 1996.