Electronic Communications of the EASST
Volume 12 (2008)

Formal Modeling of Adaptive and Mobile Processes

Analysis of Mobile Agents using Invariants of Object Nets

Michael Köhler and Daniel Moldt

18 pages

Guest Editors: Julia Padberg, Kathrin Hoffmann
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Analysis of Mobile Agents using Invariants of Object Nets

Michael Köhler and Daniel Moldt

University of Hamburg, Department for Informatics
Vogt-Kölln-Straße 30, D-22527 Hamburg
koehler@informatik.uni-hamburg.de

Abstract: Mobility induces new challenges for dynamic systems, which need a new conceptional treatment: systems, that deal for example with mobile agents, need extended security concepts to handle the risks, induced by foreign, untrusted agents.

In this contribution we use object nets to model mobile systems. Object nets are Petri nets which have Petri nets as tokens – an approach known as the nets-within-nets paradigm. Object nets are called elementary if the net system has a two levelled structure. In this work we apply structural analysis methods for object nets – namely place invariants – to a simple case study modelling mobile agents.

Keywords: linear invariants, mobility, nets-within-nets, object nets

1 Introduction

Object nets are Petri nets which have Petri nets as tokens – an approach which is called the *nets-within-nets* paradigm, proposed by Valk [Val91, Val03] for a two levelled structure and generalised in [KR03, KR04, KB09] for arbitrary nesting structures. The Petri nets that are used as tokens are called net-tokens. Net-tokens are tokens with internal structure and inner activity. This is different from place refinement, since tokens are transported while a place refinement is static. Net-tokens are some kind of *dynamic* refinement of states. Figure 1 shows an example object net with four net-tokens: two on place p_1 and one on p_2 and p_3 each. The net-tokens on p_1 and p_2 share the same net structure, but have independent markings. Object nets are useful to model the mobility of active objects or agents (cf. [KMR01] and [KMR03]).

It is quite natural to use object nets to model mobility and mobile agents. Each place of the system net describes a location that hosts agents, which are net tokens. Mobility can be modelled by moving the net token from one place to another. This hierarchy forms a useful abstraction of the system: on a high level the agent system and on a lower level of the hierarchy the agent itself.

Without the viewpoint of nets as tokens, the modeller would have to encode the agent differently, e.g. as a data-type. This has the disadvantage, that the inner actions cannot be modelled directly, so, they have to be lifted to the system net, which seems quite unnatural. By using nets-within-nets we can investigate the concurrency of the system and the agent in one model without loosing the abstraction needed.

Following [KMR03], we distinguish four different kinds of mobility, which are know as spontaneous, subjective, objective and consensual moves of the mobile agent (cf. Figure 2, where A is the agent net as a net token):

- Spontaneous Move: Neither the agent nor its environment initiate the transport. The move-

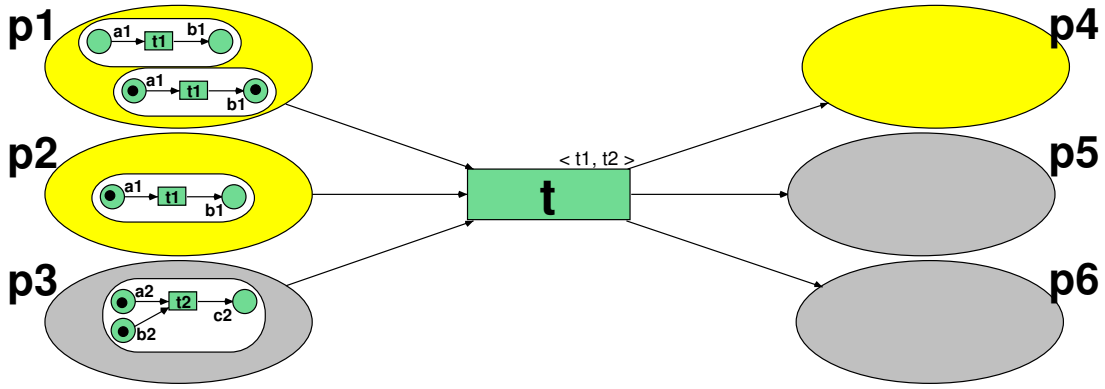


Figure 1: An Elementary Object Net

ment can take place, but it is not enforced. No coupling of the environment and the agent is needed.

- Subjective Move: The agent itself initiates the movement, so agent and environment have to be coupled. This is described by the channel move, which has to be enabled in the agent. The movement takes places if the environment is able to execute it.

- Objective Move: The environment initiates the movement of the agent. The agent is forced to be transported. The initiative of the environment is modelled by the place travel ticket.

- Consensual move: Both the environment and the agent come to an agreement on the movement. This is modelled by a combination of the channel move, which has to be enabled in the agent, and the external condition modelled by the travel ticket.

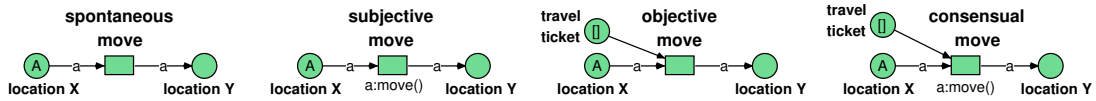


Figure 2: Types of mobility

The multi-agent architecture MULAN [KMR01] provides facilities for subjective moves as well as for objective moves.

Among the wealth of research on defining mobile systems, in recent years a variety of formalisms have been introduced or adopted to cover mobility: The approaches can be roughly separated into process calculi and Petri net based approaches. The π-calculus [MPW92], the Ambient-calculus [CGG00] and the Seal calculus [VC98] are just three of the more popular calculi. Approaches dealing with mobility and Petri nets can be found in [Val98], [Bus99], [Lom00], [XD00], [Hir02], [KMR03], [BBPP04], [Lak05], [HEM05], and [KF06].

The paper has the following structure. Section 2 recalls basic definitions. Section 3 defines elementary object systems (EOS) and presents some expressivity results. The section 4 presents the reference semantics for EOS, defined as a P/T net, and investigates the structural subclass of Generalised State Machines. Section 5 investigates the invariance calculus for EOS and demonstrates its compositionality. The paper ends with a conclusion.

2 Preliminaries

The definition of Petri nets relies on the notion of multisets. A multiset $\mathbf{m}$ on the set D is a mapping $\mathbf{m} : D \to \mathbb{N}$. Multisets are generalisations of sets in the sense that every subset of D corresponds to a multiset $\mathbf{m}$ with $\mathbf{m}(d) \leq 1$ for all $d \in D$. The notation is used for sets as well as for multisets. The meaning will be apparent from its use. Multiset addition $\mathbf{m}_1, \mathbf{m}_2 : D \to \mathbb{N}$ is defined component-wise: $(\mathbf{m}_1 + \mathbf{m}_2)(d) := \mathbf{m}_1(d) + \mathbf{m}_2(d)$. The empty multiset $\mathbf{0}$ is defined as $\mathbf{0}(d) = 0$ for all $d \in D$. Multiset-difference $\mathbf{m}_1 - \mathbf{m}_2$ is defined by $(\mathbf{m}_1 - \mathbf{m}_2)(d) := \max(\mathbf{m}_1(d) - \mathbf{m}_2(d), 0)$. We use common notations for the cardinality of a multiset $|\mathbf{m}| := \sum_{d \in D} \mathbf{m}(d)$ and multiset ordering $\mathbf{m}_1 \leq \mathbf{m}_2$ where the partial order $\leq$ is defined by $\mathbf{m}_1 \leq \mathbf{m}_2 \iff \forall d \in D : \mathbf{m}_1(d) \leq \mathbf{m}_2(d)$. A multiset $\mathbf{m}$ is finite if $|\mathbf{m}| < \infty$. The set of all finite multisets over the set D is denoted $MS(D)$. The set $MS(D)$ naturally forms a monoid with multiset addition $+$ and the empty multiset $\mathbf{0}$. Multisets can be identified with the commutative monoid structure $(MS(D), +, 0)$. Multisets are the free commutative monoid over D since every multiset has the unique representation in the form $\mathbf{m} = \sum_{d \in D} \mathbf{m}(d) \cdot d$ where $\mathbf{m}(d)$ denotes the multiplicity of d. Multisets can be represented as a formal sum in the form $\mathbf{m} = \sum_{i=1}^{|\mathbf{m}|} x_i$ where $x_i \in D$.

Any mapping $f : D \to D'$ can be extended to a homomorphism $f^\sharp : MS(D) \to MS(D')$ on multisets: $f^\sharp(\sum_{i=1}^{n} x_i) = \sum_{i=1}^{n} f(x_i)$. This includes the special case $f^\sharp(\mathbf{0}) = \mathbf{0}$. We simply write f to denote the mapping $f^\sharp$. The notation is in accordance with the set-theoretic notation $f(A) = \{f(a) | a \in A\}$.

Definition 1 A *p/t net* N is a tuple $N = (P, T, \mathbf{pre}, \mathbf{post})$, such that P is a set of places, T is a set of transitions, with $P \cap T = \emptyset$, and $\mathbf{pre}, \mathbf{post} : T \to MS(P)$ are the pre- and post-condition functions. A marking of N is a multiset of places: $\mathbf{m} \in MS(P)$. A p/t net with initial marking $\mathbf{m}$ is denoted $N = (P, T, \mathbf{pre}, \mathbf{post}, \mathbf{m})$.

We use the usual notations for nets like ${}^\bullet x$ for the set of predecessors and $x^\bullet$ for the set of successors for a node $x \in (P \cup T)$.

A transition $t \in T$ of a p/t net N is enabled in marking $\mathbf{m}$ iff $\forall p \in P : \mathbf{m}(p) \geq \mathbf{pre}(t)(p)$ holds. The successor marking when firing t is $\mathbf{m}'(p) = \mathbf{m}(p) - \mathbf{pre}(t)(p) + \mathbf{post}(t)(p)$ for all $p \in P$. Using multiset notation enabling is expressed by $\mathbf{m} \geq \mathbf{pre}(t)$ and the successor marking is $\mathbf{m}' = \mathbf{m} - \mathbf{pre}(t) + \mathbf{post}(t)$. We denote the enabling of t in marking $\mathbf{m}$ by $\mathbf{m} \xrightarrow[N]{t}$. Firing of t is denoted by $\mathbf{m} \xrightarrow[N]{t} \mathbf{m}'$. The net N is omitted if it is clear from the context.

Firing is extended to sequences $w \in T^*$ in the obvious way: (i) $\mathbf{m} \xrightarrow{\varepsilon} \mathbf{m}$; (ii) If $\mathbf{m} \xrightarrow{w} \mathbf{m}'$ and $\mathbf{m}' \xrightarrow{t} \mathbf{m}''$ hold, then we have $\mathbf{m} \xrightarrow{wt} \mathbf{m}''$.

We write $\mathbf{m} \xrightarrow{*} \mathbf{m}'$ whenver there is some $w \in T^*$ such that $\mathbf{m} \xrightarrow{w} \mathbf{m}'$ holds.

The set of reachable markings is $RS(\mathbf{m}_0) : \{\mathbf{m} \mid \exists w \in T^* : \mathbf{m}_0 \xrightarrow{w} \mathbf{m}\}$.

3 Elementary Object Systems

An elementary object system (EOS) is composed of a system net, which is a p/t net given as $\widehat{N} = (\widehat{P}, \widehat{T}, \mathbf{pre}, \mathbf{post})$ and a set of object nets $\mathscr{N} = \{N_1, \ldots, N_n\}$, which are p/t nets given as $N = (P_N, T_N, \mathbf{pre}_N, \mathbf{post}_N)$. W.l.o.g. we assume that all nodes (places and transitions) are disjoint.

The system net places are typed by the mapping $d : \widehat{P} \to \{\bullet\} \cup \mathscr{N}$ with the meaning, that the place $\widehat{p}$ of the system net contains black tokens only if $d(\widehat{p}) = \bullet$ or net-tokens of the object net type N if $d(\widehat{p}) = N$. No place of the is mapped to the system net itself.

Since the tokens of an EOS are instances of object nets a *marking* $\mu \in \mathscr{M}$ of OS is a nested multiset. The set of all markings which are syntactically consistent with the typing d is denoted $\mathscr{M}$ (Here $d^{-1}(N) \subseteq \widehat{P}$ describes the set of system net places of the type N):

$$\mathscr{M} := MS\left(\left(d^{-1}(\bullet) \times \{\mathbf{0}\}\right) \cup \bigcup_{N \in \mathscr{N}} \left(d^{-1}(N) \times MS(P_N)\right)\right) \tag{1}$$

A marking of an EOS OS is denoted $\mu = \sum_{k=1}^{|\mu|} (\widehat{p}_k, M_k)$ where $\widehat{p}_k$ is a place in the system net and M_k is the marking of the net-token of type $d(\widehat{p}_k)$. To emphasise the nesting markings are also denoted as $\mu = \sum_{k=1}^{|\mu|} \widehat{p}_k[M_k]$. Tokens of the form $\widehat{p}[\mathbf{0}]$ and $d(\widehat{p}) = \bullet$ are abbreviated as $\widehat{p}[]$.

The EOS firing rule is defined for three cases: system-autonomous firing (only a transition of the system net fires), object-autonomous firing (only a transition of an object net fires), and synchronised firing (a transition of the system net fires together with object net transitions). For the sake of uniformity of the firing rule we add the idle-transitions ε_N for each object net N where $\mathbf{pre}_N(\varepsilon_N) = \mathbf{post}_N(\varepsilon_N) = \mathbf{0}$ and the set of idle transitions $\{\varepsilon_{\widehat{p}} \mid \widehat{p} \in \widehat{P}\}$ where $\mathbf{pre}(\varepsilon_{\widehat{p}}) = \mathbf{post}(\varepsilon_{\widehat{p}}) = \widehat{p}$ for the system net. Using the idle transitions the firing rule can be reduced to the case of synchronisation (see below).

The transition labelling functions $\widehat{l}_N : \widehat{T} \to C \cup \{\varepsilon\}$ and $l_N : T_N \to C \cup \{\varepsilon\}$ define a synchronisation relation between system and object net transitions: Whenever the system net transition $\widehat{t}$ is labelled with a channel inscription, i.e. $\widehat{l}_N(\widehat{t}) \in C$ for the object net N, then $\widehat{t}$ must fire synchronously with an object net transition t of this object net N with the same label, i.e. $l_N(t) = \widehat{l}_N(\widehat{t})$. Whenever $\widehat{l}_N(\widehat{t}) = \varepsilon$ for all $N \in \mathscr{N}$, then the system net transition fires system-autonomously. Analogously, the object net transition t fires object-autonomously whenever $l_N(t) = \varepsilon$. In the graphical representation the synchronisation labelling $(\widehat{l}_N, l_N)_{N \in \mathscr{N}}$ is defined by transition inscriptions and ε is omitted.

Definition 2 (EOS) An elementary object system (EOS) is a tuple $OS = (\widehat{N}, \mathscr{N}, d, l)$ such that:

1. $\widehat{N}$ is a p/t net, called the *system net*.
2. $\mathscr{N}$ is a finite set of disjoint p/t nets, called *object nets*.
3. $d : \widehat{P} \to \{\bullet\} \cup \mathscr{N} \setminus \{\widehat{N}\}$ is a typing of the system net places.
4. $l = (\widehat{l}_N, l_N)_{N \in \mathscr{N}}$ is the synchronisation labelling.

An EOS with initial marking is a tuple $OS = (\widehat{N}, \mathscr{N}, d, l, \mu_0)$ where $\mu_0 \in \mathscr{M}$ is the initial marking of the system net.

We name special properties of EOS:

- An EOS is *minimal* iff it has exactly one type of object nets: $|\mathscr{N}| = 1$.
- An EOS is *pure* iff it has no places for black tokens: $d^{-1}(\bullet) = \emptyset$.
- An EOS is *p/t-like* iff it has only places for black tokens: $d(\widehat{P}) = \{\bullet\}$.
- An EOS is *unary* iff it is pure and minimal.
- An EOS is *monotonous* iff its typing d is. A typing is called *monotonous* iff for each place in the preset of $\widehat{t}$ typed with an object net there is place in the postset typed with the same net: $(d(\widehat{{}^\bullet t}) \cap \mathscr{N}) \subseteq (d(\widehat{t^\bullet}) \cap \mathscr{N})$.

The synchronisation labelling $(\widehat{l}_N, l_N)_{N \in \mathscr{N}}$ generates the set of system events Θ. An event is a pair $(\widehat{\tau}, \theta)$ – also denoted $\widehat{\tau}[\theta]$ in the following – where $\widehat{\tau}$ is a transition of the system net or $\varepsilon_{\widehat{p}}$ for some $\widehat{p}$ and $\theta : \mathscr{N} \to \bigcup_{N \in \mathscr{N}} (T_N \cup \{\varepsilon_N\})$ maps object nets $N \in \mathscr{N}$ to its transitions T_N including the idle transition ε_N, i.e. $\theta(N) \in (T_N \cup \{\varepsilon_N\})$. The idle map $\varepsilon_{\mathscr{N}}$ is defined $\varepsilon_{\mathscr{N}}(N) = \varepsilon_N$ for all $N \in \mathscr{N}$. We extend the labelling to idle transitions by $l_N(\varepsilon_{\widehat{p}}) = l_N(\varepsilon_N) = \varepsilon$ for all $\widehat{p} \in \widehat{P}$ and $N \in \mathscr{N}$.

$$\begin{aligned} \Theta \quad := \quad & \left\{ \widehat{t}[\theta] \mid \forall N \in \mathscr{N} : \widehat{l}_N(\widehat{t}) = l_N(\theta(N)) \right\} \\ & \cup \left\{ \varepsilon_{\widehat{p}}[\theta] \mid \exists t \in T_{d(\widehat{p})} : (\theta(\mathscr{N}) \setminus \varepsilon_{\mathscr{N}}) = \{t\} \wedge l_N(t) = \varepsilon \right\} \end{aligned} \tag{2}$$

An event $\widehat{\tau}[\theta]$ has the meaning that $\widehat{\tau}$ fires synchronously with all the object net transitions $\theta(N), N \in \mathscr{N}$. Note, that $(\varepsilon_{\widehat{p}}, \varepsilon_{\mathscr{N}})$ is excluded because it has no effect. By the construction of Θ each system net transition has exactly one synchronisation partner in each object net $N \in \mathscr{N}$. This partner might be an idle-transition. System-autonomous events have the form $(\widehat{t}, \varepsilon_{\mathscr{N}})$. For a single object-autonomous event at the location $\widehat{p}$ we have $\widehat{\tau} = \varepsilon_{\widehat{p}}$ and for all except one object net N we have $\tau_N = \varepsilon_N$, i.e. $|\theta(\mathscr{N}) \setminus \varepsilon_{\mathscr{N}}| = 1$.

Example 1 *Figure 1 shows an* EOS *with the system net* $\widehat{N}$ *and the object nets* $\mathscr{N} = \{N_1, N_2\}$. *The nets are given as* $\widehat{N} = (\widehat{P}, \widehat{T}, \mathbf{pre}, \mathbf{post})$ *with* $\widehat{P} = \{p_1, \ldots, p_6\}$ *and* $\widehat{T} = \{t\}$.

The object nets are $N_1 = (P_1, T_1, \mathbf{pre}_1, \mathbf{post}_1)$ *with* $P_1 = \{a_1, b_1\}$ *and* $T_1 = \{t_1\}$ *and* $N_2 = (P_2, T_2, \mathbf{pre}_2, \mathbf{post}_2)$ *with* $P_2 = \{a_2, b_2, c_2\}$ *and* $T_2 = \{t_2\}$.

The typing is $d(p_1) = d(p_2) = d(p_4) = N_1$ *and* $d(p_3) = d(p_5) = d(p_6) = N_2$. *The typing is illustrated in Figure 1 by different colours for the places.*

There is only one synchronous event: $\Theta = \{t[N_1 \mapsto t_1, N_2 \mapsto t_2]\}$. *If there is at most one transition for each label* $c \in C$ *then we can simply denote the corresponding transitions, like in the inscription* $\langle t_1, t_2 \rangle$ *at transition* t *in Figure 1.*

The initial marking has two net-tokens on p_1, *one on* p_2, *and one on* p_3:

$$\mu = p_1[a_1 + b_1] + p_1[\mathbf{0}] + p_2[a_1] + p_3[a_2 + b_2]$$

Note, that for Figure 1 the structure is the same for the three net-tokens on p_1 *and* p_2 *but the net-tokens' markings are different.*

3.1 Projections and Firing Rule

Let $\mu = \sum_{k=1}^{|\mu|}(\widehat{p}_k, M_k)$ be a marking of an EOS. The projection Π^1 on the first component abstracts away the substructure of all net-tokens:

$$\Pi^1\left(\sum_{k=1}^{|\mu|}\widehat{p}_k[M_k]\right) := \sum_{k=1}^{|\mu|}\widehat{p}_k \tag{3}$$

The projection Π^2_N on the second component is the abstract marking of all net-tokens of the type $N \in \mathscr{N}$ ignoring their local distribution within the system net.

$$\Pi^2_N\left(\sum_{k=1}^{|\mu|}\widehat{p}_k[M_k]\right) := \sum_{k=1}^{|\mu|}\mathbf{1}_N(\widehat{p}_k)\cdot M_k \tag{4}$$

where the indicator function $\mathbf{1}_N : \widehat{P} \to \{0,1\}$ is $\mathbf{1}_N(\widehat{p}) = 1$ iff $d(\widehat{p}) = N$. Note, that $\Pi^2_N(\mu)$ results in an marking of the object net N.

A system event $\widehat{\tau}[\theta]$ removes net-tokens together with their individual internal markings. Firing the event replaces a nested multiset $\lambda \in \mathscr{M}$ that is part of the current marking μ, i.e. $\lambda \leq \mu$, by the nested multiset ρ. The enabling condition is expressed by the *enabling predicate* ϕ_{OS} (or just ϕ whenever OS is clear from the context):

$$\begin{aligned}\phi(\widehat{\tau}[\theta],\lambda,\rho) \quad \Longleftrightarrow\ & \Pi^1(\lambda) = \mathbf{pre}(\widehat{\tau}) \wedge \Pi^1(\rho) = \mathbf{post}(\widehat{\tau}) \wedge \\ & \forall N \in \mathscr{N} : \Pi^2_N(\lambda) \geq \mathbf{pre}_N(\theta(N)) \wedge \\ & \forall N \in \mathscr{N} : \Pi^2_N(\rho) = \Pi^2_N(\lambda) - \mathbf{pre}_N(\theta(N)) + \mathbf{post}_N(\theta(N))\end{aligned} \tag{5}$$

With $\widehat{M} = \Pi^1(\lambda)$ and $\widehat{M}' = \Pi^1(\rho)$ as well as $M_N = \Pi^2_N(\lambda)$ and $M'_N = \Pi^2_N(\rho)$ for all $N \in \mathscr{N}$ the predicate ϕ has the following meaning:

1. The first conjunct expresses that the system net multiset $\widehat{M}$ corresponds to the pre-condition of the system net transition $\widehat{t}$, i.e. $\widehat{M} = \mathbf{pre}(\widehat{t})$.
2. In turn, a multiset $\widehat{M}'$ is produced, that corresponds with the post-set of $\widehat{t}$.
3. An object net transition τ_N is enabled if the combination M_N of the markings net-tokens of type N enable it, i.e. $M_N \geq \mathbf{pre}_N(\theta(N))$.
4. The firing of $\widehat{\tau}[\theta]$ must also obey the *object marking distribution condition* $M'_N = M_N - \mathbf{pre}_N(\theta(N)) + \mathbf{post}_N(\theta(N))$ where $\mathbf{post}_N(\theta(N)) - \mathbf{pre}_N(\theta(N))$ is the effect of the object net's transition on the net-tokens.

Note, that (1) and (2) assures that only net-tokens relevant for the firing are included in λ and ρ. Conditions (3) and (4) allows for additonal tokens in the net-tokens.

For system-autonomous events $\widehat{t}[\varepsilon_{\mathscr{N}}]$ the enabling predicate ϕ can be simplified further. We have $\mathbf{pre}_N(\varepsilon_N) = \mathbf{post}_N(\varepsilon_N) = \mathbf{0}$. This ensures $\Pi^2_N(\lambda) = \Pi^2_N(\rho)$, i.e. the sum of markings in the copies of a net-token is preserved w.r.t. each type N. This condition ensures the existence of linear invariance properties (cf. Theorem 5).

Analogously, for an object-autonomous event we have an idle-transition $\widehat{\tau} = \varepsilon_{\widehat{p}}$ for the system net and the first and the second conjunct is: $\Pi^1(\lambda) = \mathbf{pre}(\widehat{t}) = \widehat{p} = \mathbf{post}(\widehat{t}) = \Pi^1(\rho)$. So, there is an addend $\lambda = \widehat{p}[M]$ in μ with $d(\widehat{p}) = N$ and M enables $t_N := \theta(N)$.

Definition 3 (Firing Rule) Let OS be an EOS and $\mu, \mu' \in \mathscr{M}$ markings. The event $\widehat{\tau}[\theta]$ is enabled in μ for the mode $(\lambda, \rho) \in \mathscr{M}^2$ iff $\lambda \leq \mu \wedge \phi(\widehat{\tau}[\theta], \lambda, \rho)$ holds.

An event $\widehat{\tau}[\theta]$ that is enabled in μ for the mode (λ, ρ) can fire: $\mu \xrightarrow[OS]{\widehat{\tau}[\theta](\lambda,\rho)} \mu'$. The resulting successor marking is defined as $\mu' = \mu - \lambda + \rho$.

We write $\mu \xrightarrow[OS]{\widehat{\tau}[\theta]} \mu'$ whenever $\mu \xrightarrow[OS]{\widehat{\tau}[\theta](\lambda,\rho)} \mu'$ for some mode (λ, ρ).

Example 2 *Consider the* EOS *of Figure 1 again. The current marking μ of the* EOS *enables $t[N_1 \mapsto t_1, N_2 \mapsto t_2]$ in the mode (λ, ρ) where*

$$\begin{aligned} \lambda &= p_1[a_1 + b_1] + p_2[a_1] + p_3[a_2 + b_2] \\ \rho &= p_4[a_1 + b_1 + b_1] + p_5[\mathbf{0}] + p_6[c_2] \end{aligned}$$

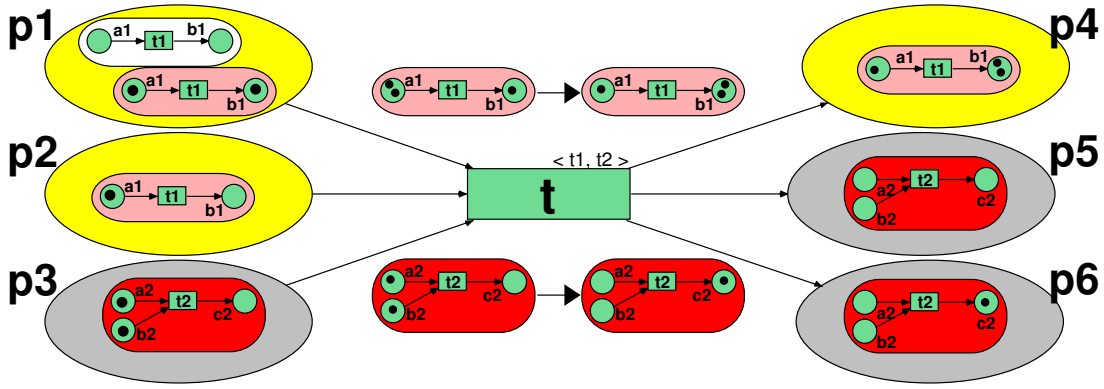


Figure 3: The EOS of Figure 1 after the firing of $t[N_1 \mapsto t_1, N_2 \mapsto t_2]$

The selected net-tokens of λ are highlighted in Figure 3 (Ignore the tokens on p_4, p_5, and p_6 for the moment.). We have the current marking:

$$\mu = p_1[\mathbf{0}] + \underbrace{p_1[a_1 + b_1] + p_2[a_1] + p_3[a_2 + b_2]}_{\lambda} = p_1[\mathbf{0}] + \lambda$$

The net-tokens' markings are added. The sub-synchronisation are shown above and below the transition t. After the synchronisation we obtain the successor marking on p_4, p_5, and p_6 as shown in the Figure 3:

$$\begin{aligned} \mu' &= (\mu - \lambda) + \rho = p_1[\mathbf{0}] + \rho \\ &= p_1[\mathbf{0}] + p_4[a_1 + b_1 + b_1] + p_5[\mathbf{0}] + p_6[c_2] \end{aligned}$$

EOS are a canonical extension of p/t nets in two ways: The behaviour of the system net in the EOS when ignoring the net-tokens structure cannot be distinguished from the system net as a p/t net (Lemma 1) and each p/t-like EOS is isomorphic to the system net as a p/t net (Lemma 2) .

EOS are a canonical extension of p/t nets, since the behaviour of an EOS when considering only the system net's perspective is in accordance with the behaviour of the system net considered as a p/t net, i.e. if an event $\widehat{t}$ is disabled in the p/t net then for all θ the event $\widehat{t}[\theta]$ is disabled in the EOS.

Lemma 1 *For $OS = (\widehat{N}, \mathscr{N}, d, l, \mu_0)$ define $\Pi^1(OS) = \widehat{N}$. For each* EOS *OS we have:*

$$\mu \xrightarrow[OS]{\widehat{t}[\theta]} \mu' \Longrightarrow \Pi^1(\mu) \xrightarrow[\Pi^1(OS)]{\widehat{t}} \Pi^1(\mu')$$

Proof. First, we have that $\Pi^1(\mu)$ is a marking of the p/t net $\widehat{N}$. Whenever μ enables $\widehat{t}[\theta]$ for a mode (λ, ρ) then $\phi(\widehat{t}[\theta], \lambda, \rho)$ holds which implies $\Pi^1(\lambda) = \mathbf{pre}(\widehat{t})$ and $\Pi^1(\rho) = \mathbf{post}(\widehat{t})$ and $\mu' = \mu - \lambda + \rho$.

Since $\mu \geq \lambda$ we have $\Pi^1(\lambda) \geq \Pi^1(\lambda) = \mathbf{pre}(\widehat{t})$, i.e. $\widehat{t}$ is enabled in $\Pi^1(\lambda)$.

For the system net projection follows:

$$\Pi^1(\mu') = \Pi^1(\mu - \lambda + \rho) = \Pi^1(\mu) - \Pi^1(\lambda) + \Pi^1(\rho) = \Pi^1(\mu) - \mathbf{pre}(\widehat{t}) + \mathbf{post}(\widehat{t})$$

This is the successor marking when firing $\widehat{t}$ in $\Pi^1(\mu)$ for the p/t net $\widehat{N}$. □

For a p/t-like EOS we have no object nets: $\mathscr{N} = \emptyset$, synchronisation given as $\Theta = \{\widehat{t}[\emptyset] \mid \widehat{t} \in \widehat{T}\}$, and the typing is the constant function $d = \bullet$ with $\bullet(\widehat{p}) = \bullet$ for all $\widehat{p} \in \widehat{P}$. The initial marking contains no submarking: $\mu_0 \in \widehat{P} \times \{\mathbf{0}\} \subseteq \mathscr{M}$. So, p/t-like EOS have the form:

$$OS = (\widehat{N}, \emptyset, \bullet, \{\widehat{t}[\emptyset] \mid \widehat{t} \in \widehat{T}\}, \mu_0)$$

Lemma 2 *A p/t-like* EOS *$OS = (\widehat{N}, \emptyset, \bullet, l, \mu_0)$ is isomorphic to the p/t net $(\widehat{N}, \Pi^1(\mu_0))$:*

$$\mu \xrightarrow[OS]{(\widehat{\tau},\emptyset)(\lambda,\rho)} \mu' \iff \Pi^1(\mu) \xrightarrow[\widehat{N}]{\widehat{\tau}} \Pi^1(\mu')$$

Proof. For a p/t-like EOS the predicate $\phi(\widehat{\tau}[\theta], \lambda, \rho)$ reduces to $\Pi^1(\lambda) = \mathbf{pre}(\widehat{\tau}) \wedge \Pi^1(\rho) = \mathbf{post}(\widehat{\tau})$ since $\mathscr{N} = \emptyset$. Therefore $\Pi^2(\mu) = \mathbf{0}$ holds for all reachable markings μ.

Since $\lambda \leq \mu$ we have $\Pi^1(\lambda) \leq \Pi^1(\mu)$ where $\Pi^1(\mu)$ is the marking in the p/t net $\widehat{N}$. The successor marking when firing $\widehat{\tau}[\emptyset](\lambda, \rho)$ in OS is defined as $\mu' = \mu - \lambda + \rho$. Obviously, $\Pi^2_N(\mu') = \mathbf{0}$ and $\Pi^1(\mu') = \Pi^1(\mu) - \mathbf{pre}(\widehat{\tau}) + \mathbf{post}(\widehat{\tau})$ which equals the successor marking when firing $\widehat{t}$ in $\widehat{N}$. □

3.2 Mobile EOS and Marking Equivalence

We define the relation $\cong \subseteq \mathscr{M}^2$ on nested multisets, that relates nested markings which coincide in their projections. The *projection equivalence* $\cong$ is a relation on $\mathscr{M}$ defined by:

$$\alpha \cong \beta \quad :\Longleftrightarrow \quad \Pi^1(\alpha) = \Pi^1(\beta) \wedge \forall N \in \mathscr{N} : \Pi^2_N(\alpha) = \Pi^2_N(\beta) \tag{6}$$

Obviously, there are several markings μ with the same projection, i.e. μ is not uniquely defined by $\Pi(\mu)$. Defining the projection of a marking μ as

$$\Pi(\mu) := (\Pi^1(\mu), (\Pi^2_N(\mu))_{N \in \mathscr{N}}) \tag{7}$$

Projection equivalence $\alpha \cong \beta$ holds if and only if $\Pi(\alpha) = \Pi(\beta)$. The projection $\Pi(\mu)$ is an representant of the equivalence class $[\mu]_{\cong}$.

The relation $\alpha \cong \beta$ abstracts from the location, i.e. the concrete net-token, in which a object net's place p is marked as long as it is present in α and β. For example, for $d(\widehat{p}) = d(\widehat{p}')$ we have

$$\widehat{p}[p_1 + p_2] + \widehat{p}'[p_3] \cong \widehat{p}[p_3 + p_2] + \widehat{p}'[p_1]$$

which means that $\cong$ allows the tokens p_1 and p_3 to change their locations (i.e. between $\widehat{p}$ and $\widehat{p}'$).

Lemma 3 *The enabling predicate is invariant with respect to the relation $\cong$:*

$$\phi(\widehat{\tau}[\theta], \lambda, \rho) \iff (\forall \lambda', \rho' : \lambda' \cong \lambda \wedge \rho' \cong \rho \Longrightarrow \phi(\widehat{\tau}[\theta], \lambda', \rho'))$$

Proof. From the definition of ϕ one can see that the firing mode (λ, ρ) is used only via its projection by Π. Since $\lambda' \cong \lambda, \rho' \cong \rho$ expresses equality modulo projection the predicate ϕ cannot distinguish between λ' and λ, resp. ρ' and ρ. □

A note on the monotonicity of the typing d: A transition $\widehat{t} \in \widehat{T}$ with an object net N that is present in the postset, but not in the preset (i.e. $N \notin d({}^\bullet\widehat{t})$ and $N \in d(\widehat{t}^\bullet)$) generates net-tokens of type N. The firing rule ensures that these net-tokens carry the empty marking since in this case $(\widehat{\tau}, C)$ is enabled in mode (λ, ρ) only if all object nets in ρ of this type N carry the empty marking.

The symmetric case, i.e. $N \in d({}^\bullet\widehat{t})$ and $N \notin d(\widehat{t}^\bullet)$, which destroys net-tokens of type N, is forbidden by the monotonous typing, since it is problematic: In this case $(\widehat{\tau}, C)$ is enabled in mode (λ, ρ) only if all object nets in λ of this type N carry the empty marking: $\Pi^2_N(\lambda) = \mathbf{0}$. So, not all pairs (λ', ρ') with $\lambda \preceq \lambda'$ (i.e. more tokens in the net-tokens) are also firing modes, i.e. the firing rule would *not* be monotonous.

In [KR04] we have shown that the nesting structure of object net markings (which allow an arbitrary deep nesting) can be used to encode counter automatons. This technique cannot applied to elementary object nets, because they are restricted to a two levelled structure. Nevertheless, elementary object nets are very powerful. The interesting part in the firing rule of EOS is the fact that moving an object net token in the system net has the power to modify the state of an unbounded number of tokens, i.e. all the tokens of the object net tokens (including the case of zero tokens). In [Köh07, Theorem 3.1] we have investigated the consequences for the reachability problem.

Theorem 1 *Reachability is undecidable for non-minimal, pure* EOS *and for minimal, non-pure* EOS.

Due to the monotonos assumption for the typing d of EOS boundedness remains decidable for EOS [Köh06, Theorem 7].

3.3 Mobile Object Nets

Projection equivalence $\cong$ abstracts from the concrete place in the preset of an event. In [KF07] we replaced the condition $\forall N \in \mathcal{N} : \Pi^2_N(\alpha) = \Pi^2_N(\beta)$ of (6) by the more general one using an equivalence $\leftrightarrow$ on the object nets' places:

$$\alpha \leftrightarrow \beta \iff \Pi^1(\alpha) = \Pi^1(\beta) \wedge \forall N \in \mathcal{N} : \Pi^2_N(\alpha) = \Pi^2_N(\beta) \pmod{\leftrightarrow} \tag{8}$$

Iff we modify the firing rule given in Definition 3 using the equivalence $\leftrightarrow$ instead of $\cong$ we obtain the formalism of *mobile object nets* (cf. [KF07]) In Figure 4 the infrastructure is composed of the two buildings A and B represented in the system net. Buildings can be seen as a metaphor for namespaces, e.g., for different hosts on a distributed network or different WLAN areas. The two buildings are connected via the mobility transfer transitions t_4 and t_6. One mobile agent is present inside building A as a net token. Inside the building the agent has access to a workflow describing how the agent is allowed to use services, i.e. the building's infrastructure. The agent can decide to use the building's infrastructure by synchronising with the access workflow's transitions.

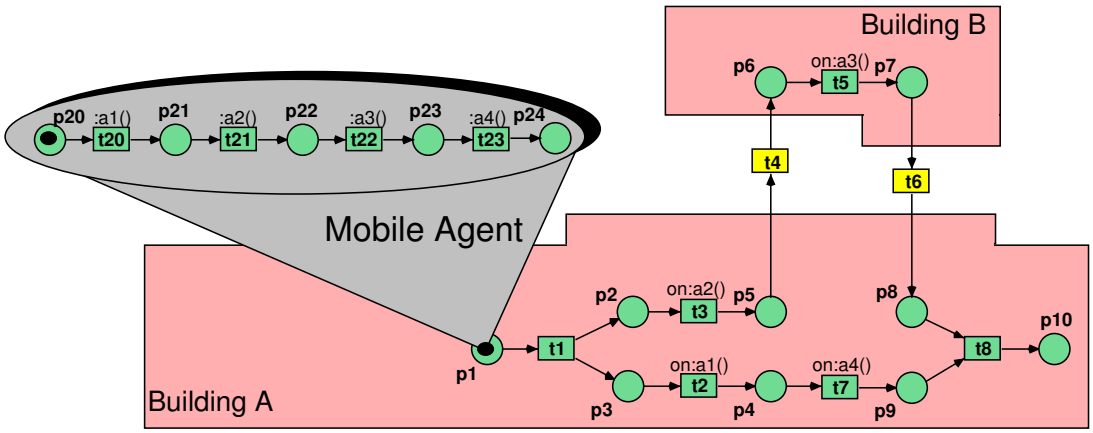


Figure 4: A mobile agent's environment

When modelling this scenario we have to distinguish two kinds of movement: Movement within a building and movement from one building to another. When moving whithin a building, the agent has full access to all services (e.g. service stations, information servers, etc.). On the other hand, when moving to a different building the environment may change dramatically: Services may become unavailable, they may change their name or their kind of access protocols. This leads to the usual problem that within the same environment (e.g. the memory of a personal computer) we can use pointers to access objects (as done for Java objects), which is obviously impossible for a distributed space like a computer network: For example when a Java program transfers an object from machine A to B via remote method invocation (RMI) it does not transfer the object's pointers (which are not valid for B); instead Java rather makes a *deep copy* of the object (called *serialisation*) and transfers this value over the network. The value is used to generate a new object at B which can be accessed by a fresh pointer.

The formalism of mobile object nets is well suited to this finer granularity of namespaces while elementary objects are not since for object nets each place is one single namespace. The expressibility of mobile object nets leads to the fact that mobile object net can simulate Petri nets with inhibitor arcs (cf. Theorem 6 in [KF07]) which proves that mobile EOS are more powerful than EOS since boundedness is undecidable for inhibitor nets, but decidable for EOS.

4 Reference Semantics and Generalised State Machines

For each EOS there is an obvious construction of a p/t net, called the reference net, which is constructed by taking as the set of places the disjoint union of all places and as the set of transitions the synchronisations. Since the places of all nets in $\mathscr{N}$ are disjoint by definition, the decomposition $(\Pi^1(\mu),(\Pi^2_N(\mu))_{N\in\mathscr{N}})$ can be identified with the mixed multiset:

$$\Pi^1(\mu)+\sum_{N\in\mathscr{N}}\Pi^2_N(\mu)$$

This sum is another representant of the equivalence class $[\mu]_{\cong}$.

Definition 4 Let $OS=(\widehat{N},\mathscr{N},d,l,\mu_0)$ be an EOS. The *reference net* $\mathrm{RN}(OS)$ is defined as the p/t net:

$$\mathrm{RN}(OS)=\left(\left(\widehat{P}\cup\bigcup_{N\in\mathscr{N}}P_N\right),\Theta,\mathbf{pre}^{\mathrm{RN}},\mathbf{post}^{\mathrm{RN}},\mathrm{RN}(\mu_0)\right)$$

where $\mathbf{pre}^{\mathrm{RN}}$ (and analogously $\mathbf{post}^{\mathrm{RN}}$) is defined by:

$$\mathbf{pre}^{\mathrm{RN}}(\widehat{\tau}[\theta])=\mathbf{pre}(\widehat{\tau})+\sum_{N\in\mathscr{N}}\mathbf{pre}_N(\theta(N))$$

and for markings we define:

$$\mathrm{RN}(\mu):=\Pi^1(\mu)+\sum_{N\in\mathscr{N}}\Pi^2_N(\mu)$$

The net is called *reference net* because it behaves as if each object net would have been accessed via pointers and not like a value: A black token on a system net place $\widehat{p}$ is interpreted as a pointer to the object $\widehat{N}=d(\widehat{p})$ where each object net has exactly one instance but several pointers referring to it.

Theorem 2 *Let OS be an* EOS. *Every event* $\widehat{\tau}[\theta]$ *that is activated in OS for* (λ,ρ) *is so in* $\mathrm{RN}(OS)$*:*

$$\mu\xrightarrow[OS]{\widehat{\tau}[\theta](\lambda,\rho)}\mu'\quad\Longrightarrow\quad\mathrm{RN}(\mu)\xrightarrow[\mathrm{RN}(OS)]{\widehat{\tau}[\theta]}\mathrm{RN}(\mu')$$

Proof. Whenever $\widehat{\tau}[\theta]$ is activated in μ the enabling condition ϕ holds. This implies that $\Pi^1(\mu)$ enables $\widehat{\tau}$ and $\Pi^2_N(\mu)$ enables $\theta(N)$ for each $N\in\mathscr{N}$. Since all the places are disjoint $\mathrm{RN}(\mu)$ is isomorphic to the projections $\Pi(\mu)$ and this implies that the multiset sum $\widehat{\tau}+\sum_{N\in\mathscr{N}}\theta(N)$ is enabled which is equivalent to the enabling in $\mathrm{RN}(OS)$. Analogously one can observe that the effect on $\Pi^1(\mu)$ an on the $\Pi^2_N(\mu)$ is the same which implies that the successor marking in $\mathrm{RN}(OS)$ is $\mathrm{RN}(\mu')$. □

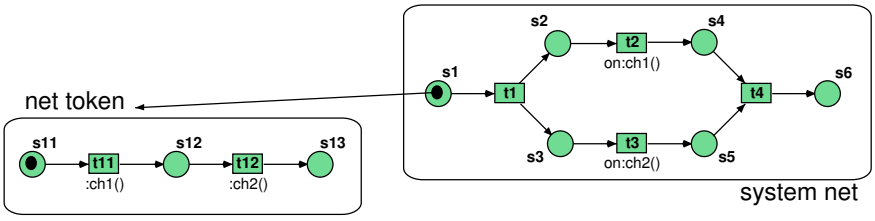


Figure 5: A sample EOS

The converse is not true in general, which can be demonstrated using the EOS in Fig. 5 known as the α-centauri example, cf. [Val98]. Initially we have $\mu_0 = \widehat{s_1}[s_{11}]$. In the reference net we have the initial marking $\text{RN}(\mu_0) = \widehat{s_1} + s_{11}$ which activates the firing sequence:

$$(\widehat{s_1} + s_{11}) \xrightarrow{\widehat{t_1}[\varepsilon]} (\widehat{s_2} + \widehat{s_3} + s_{11}) \xrightarrow{\widehat{t_2}[t_{11}]} (\widehat{s_4} + \widehat{s_3} + s_{12}) \xrightarrow{\widehat{t_3}[t_{12}]} (\widehat{s_4} + \widehat{s_5} + s_{13})$$

It is easy to see that in the EOS we can fire only a prefix, depending on the choice of the modes. The first mode assigns the token on s_{11} to the net-token on $\widehat{s_3}$:

$$\widehat{s_1}[s_{11}] \xrightarrow{\widehat{t_1}[\varepsilon]} \widehat{s_2}[\mathbf{0}] + \widehat{s_3}[s_{11}]$$

The second mode assigns the token on s_{11} to the net-token on $\widehat{s_2}$:

$$\widehat{s_1}[s_{11}] \xrightarrow{\widehat{t_1}[\varepsilon]} \widehat{s_2}[s_{11}] + \widehat{s_3}[\mathbf{0}] \xrightarrow{\widehat{t_2}[t_{11}]} \widehat{s_4}[s_{12}] + \widehat{s_3}[\mathbf{0}]$$

Since the effect in the object net is only local, $\widehat{t_3}[t_{12}]$ is not activated. So $w = \widehat{t_1}[\varepsilon] \cdot \widehat{t_2}[t_{11}] \cdot \widehat{t_3}[t_{12}]$ is a possible firing sequence for the reference net, but not for the object net system.

From Theorem 2 and the above the following property follows.

Corollary 1 *Let OS be an* EOS. *If μ is reachable from μ_0, then* $\text{RN}(\mu)$ *is reachable from* $\text{RN}(\mu_0)$. *The reverse does not hold in general.*

So, we obtain only a sufficient condition for non-reachability: The marking μ is not reachable from μ_0 whenever $\text{RN}(\mu)$ is not reachable from $\text{RN}(\mu_0)$.

Fortunately, many practical models are *Generalised State Machines* and this sufficient condition can be strengthened to a necessary one for these. An EOS *OS* is a generalised state machine iff for all $\widehat{t}$ there is either exactly one place in the preset and one in the postset typed with the object net N or there are no such places:

$$\forall N \in \mathcal{N} : \forall \widehat{t} \in \widehat{T} : \left|\{\widehat{p} \in {}^\bullet\widehat{t} \mid d(\widehat{p}) = N\}\right| = \left|\{\widehat{p} \in \widehat{t}^\bullet \mid d(\widehat{p}) = N\}\right| \leq 1 \tag{9}$$

and the inital marking has at most one net-token of each type:

$$\forall N \in \mathcal{N} : |\{\widehat{p} \in \widehat{P} \mid \Pi^1(\mu)(\widehat{p}) > 0 \wedge d(\widehat{p}) = N\}| \leq 1 \tag{10}$$

Obviously every p/t like EOS is a generalised state machine since $d(\widehat{p}) = \bullet$ for all $\widehat{p}$. In addition generalised state machines are monotonous.

For generalised state machines we can strengthen Theorem 2.

Theorem 3 *Let OS be an* EOS *with the generalised state machine property.*
A transition $\widehat{\tau}[\theta]$ *is activated in OS for* (λ,ρ) *iff it is in* $\mathrm{RN}(OS)$*:*

$$\mu \xrightarrow[OS]{\widehat{\tau}[\theta](\lambda,\rho)} \mu' \quad \Longleftrightarrow \quad \mathrm{RN}(\mu) \xrightarrow[\mathrm{RN}(OS)]{\widehat{\tau}[\theta]} \mathrm{RN}(\mu')$$

Proof. Define the property ψ as follows:

$$\psi_{gsm}(\mu) \quad = \quad \forall N \in \mathscr{N} : |\{\widehat{p} \in \widehat{P} \mid \Pi^1(\mu)(\widehat{p}) > 0 \wedge d(\widehat{p}) = N\}| \leq 1$$

By (10) the property holds initially, i.e. $\psi_{gsm}(\mu_0)$ is true. It is easy to observe that that the property $\psi_{gsm}(\mu)$ remains true in all reachable markings, since whenever there is at most one net-token for each object net in marking μ, then (9) implies that there are equally many in the successor marking μ'.

Therefore in each reachable marking μ we have for each object net N that is present in the initial marking exactly one marked system net place $\widehat{p}_N$ which contains the net-token of type N.

In this case all tokens in the projection $\Pi^2_N(\mu)$ belong to the marking of the net-token on $\widehat{p}_N$. The net-token can be reconstructed as $\widehat{p}_N[\Pi^2_N(\mu)]$.

Therefor, we can uniquely reconstruct μ from $\mathrm{RN}(\mu)$ and reachability in the net $\mathrm{RN}(OS)$ is a necessary and sufficient condition for reachability. □

A generalised state machine *OS* is therefor isomorphic with its reference net $\mathrm{RN}(OS)$.

Note that the formalism defined in [BBPP04] is restricted to generalised state machines – the general case is not considered which simplifies notations considerably but limits the expressiveness.

5 The Invariance Calculus for Object Nets

There is a well elaborated connection of Petri nets and linear algebra (cf. [Lau87, STC98]). Let $\Delta : T \to (P \to \mathbb{Z})$ be the function defined by:

$$\Delta(t)(p) = \mathbf{post}(t)(p) - \mathbf{pre}(t)(p)$$

$\Delta(t)$ denotes the *effect* of firing t. The function Δ is linear, in the sense that the effect $\Delta(t_1+t_2)$ of a transition multiset is their cumulated effect:

$$\Delta(t_1+t_2) = \Delta(t_1)+\Delta(t_2)$$

If $0 < |P|,|T| < \infty$ then Δ can be expressed as a $|P| \cdot |T|$ matrix (called *incidence matrix*) defined by $\Delta(p,t) = \mathbf{post}(t)(p) - \mathbf{pre}(t)(p)$. Using $\Delta(t)$ the firing step $\mathbf{m} \xrightarrow{t} \mathbf{m}'$ of a p/t net can be expressed as:

$$\mathbf{m}' = \mathbf{m} - \mathbf{pre}(t) + \mathbf{post}(t) = \mathbf{m} + \Delta(t)$$

It is well known that all solutions $\mathbf{i} \in \mathbb{Z}^{|P|} \setminus \{\mathbf{0}\}$ of the equation

$$\Delta^\top \mathbf{i} = \mathbf{0}$$

which are called place-invariants (short: *P*-invariants) result into a linear equation for all reachable markings $\mathbf{m} \in RS(N,\mathbf{m}_0)$.

Theorem 4 (Lautenbach) *Let* $\mathbf{i} \in \mathbb{Z}^{|P|}$ *be a P-invariant of the p/t net N. Then we have:*

$$\forall \mathbf{m} \in RS(N, \mathbf{m}_0) : \mathbf{i} \cdot \mathbf{m} = \mathbf{i} \cdot \mathbf{m}_0$$

This invariance calculus for p/t nets can be extended to EOS in a compositional way, i.e. invariance equations can be obtained from the invariance equations of the constituting components separately.

Theorem 5 *Let* $OS = (\widehat{N}, \mathscr{N}, d, l, \mu_0)$ *be an* EOS, $\widehat{\mathbf{i}}$ *a P-invariant of the system net* $\widehat{N}$ *and* $\mathbf{i}_N$ *one for each object net* $N \in \mathscr{N}$. *For all reachable markings* μ *it holds:*

$$\begin{aligned} \widehat{\mathbf{i}} \cdot \Pi^1(\mu) &= \widehat{\mathbf{i}} \cdot \Pi^1(\mu_0) \\ \forall N \in \mathscr{N} : \mathbf{i}_N \cdot \Pi^2_N(\mu) &= \mathbf{i}_N \cdot \Pi^2_N(\mu_0) \end{aligned}$$

Proof. Proof by induction on the length of the firing sequence. Induction base: For the empty sequence we have $\mu = \mu_0$ and the property is obvious.

Induction step: Assume we have $\mu_0 \xrightarrow[OS]{*} \mu \xrightarrow[OS]{\widehat{t}[\theta](\lambda,\rho)} \mu'$. Since $\widehat{\mathbf{i}}$ is an invariant of the system net we have $\widehat{\mathbf{i}} \cdot (\mathbf{post} - \mathbf{pre}) = \mathbf{0}$. It follows:

$$\begin{aligned} \widehat{\mathbf{i}} \cdot \Pi^1(\mu') &= \widehat{\mathbf{i}} \cdot \Pi^1(\mu - \lambda + \rho) = \widehat{\mathbf{i}} \cdot (\Pi^1(\mu) - \Pi^1(\lambda) + \Pi^1(\rho)) \\ &= \widehat{\mathbf{i}} \cdot \Pi^1(\mu) - \widehat{\mathbf{i}} \cdot \mathbf{pre}(\widehat{t}) + \widehat{\mathbf{i}} \cdot \mathbf{post}(\widehat{t}) = \widehat{\mathbf{i}} \cdot \Pi^1(\mu) \end{aligned}$$

For all $N \in \mathscr{N}$ we have $\mathbf{i}_N \cdot (\mathbf{post}_N - \mathbf{pre}_N) = \mathbf{0}$. It follows:

$$\mathbf{i}_N \cdot \Pi^2_N(\rho) = \mathbf{i}_N \cdot \left(\Pi^2_N(\lambda) - \mathbf{pre}_N(\theta(N)) + \mathbf{post}_N(\theta(N))\right) = \mathbf{i}_N \cdot \Pi^2_N(\lambda)$$

This proves the property. □

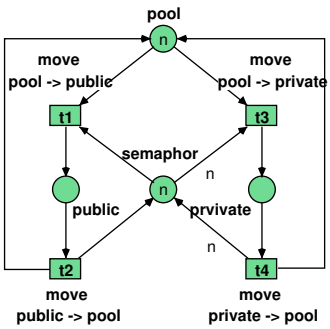
This extension of linear invariants to EOS shows that safety properties of object nets – when considered as p/t nets – are conservatively embedded. Of course, this embedding does not extend to liveness properties, since e.g. a deadlock-free system net may block when embedded into an EOS, simply because it may be synchronised with a deadlocked object net.

Example 3 *As mentioned in the introduction structural analysis is useful for the system's as well as for the mobile agent's side. The mobility infrastructure given in Fig. 6 consists of the three localities* pool, public, *and* private. *The net itself is a variant of the reader/writer problem. The parameter* $n \in \mathbb{N}$ *denotes the capacity of the public location.*

In the first step only the agent system (i.e. the system net $\widehat{N}$*) is shown, since an agent (i.e. the object net N) cannot be restricted by a platform in advance. The synchronisation relation is also omitted for the same reason.*

We show how invariants of the system net extend towards properties of the whole EOS. *The following analysis holds for arbitrarily structured agents.*

There are three locations: pool, public, *and* private. *The pool location is the initialisation area; the public area is open for any agent, while the private area has restricted access: It is allowed that many agents are simultaneously in the public location, but there can be at most*



$$\Delta = \begin{array}{|c|cccc|} \hline & t_1 & t_2 & t_3 & t_4 \\ \hline pool & -1 & 1 & -1 & 1 \\ public & 1 & -1 & & \\ semaphor & -1 & 1 & -n & n \\ private & & & 1 & -1 \\ \hline \end{array}$$

Figure 6: The Multi-Agent System Net $\widehat{N}$

one agent in the private location. This prevents agents from being spied out. The transitions between the locations model movement, which are either objective or consensual (depending on the synchronisation relation).

In the following the system net $\widehat{N}$ is analysed using invariants.

We obtain $\widehat{\mathbf{i}} = (0,1,1,n)'$ as a solution of the equation $\widehat{\mathbf{i}} \cdot \Delta = \mathbf{0}$. Using Theorem 5 we have $\widehat{\mathbf{i}}_1 \cdot \Pi^1(\mu) = \widehat{\mathbf{i}}_1 \cdot \Pi^1(\mu_0)$ for all reachable markings μ:

$$\widehat{\mathbf{i}}_1 \cdot \Pi^1(\mu) = \Pi^1(\mu)(public) + \Pi^1(\mu)(semaphor) + n \cdot \Pi^1(\mu)(private) = \widehat{\mathbf{i}}_1 \cdot \Pi^1(\mu_0) = n$$

Therefore $\Pi^1(\mu)(private) > 0$ implies $\Pi^1(\mu)(private) = 1$ and $\Pi^1(\mu)(public) = 0$.

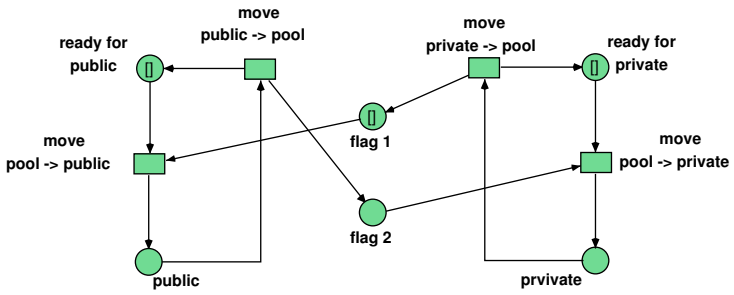


Figure 7: The Agent

In the following we analyse the agent net $\widehat{N}$ given in Fig. 7. The two places $flag_1$ and $flag_2$ are used to toggle the agent's choice between the public and the private place. The incidence matrix

is given as (with slight abbreviations):

$\Delta =$

	$pool \to pub$	$pub \to pool$	$pool \to prv$	$prv \to pool$
ready public	-1	1		
public	1	-1		
ready private			-1	1
private			1	-1
$flag_1$	-1			1
$flag_2$		1	-1	

Solving the equation $\mathbf{i} \cdot \Delta = \mathbf{0}$ *we obtain* $\mathbf{i} = (0,1,0,1,1,1)'$ *as an invariant of the agent-net. Using Theorem 5 we have* $\mathbf{i}_2 \cdot \Pi^2(\mu) = \mathbf{i}_1 \cdot \Pi^2(\mu_0)$ *for all reachable markings* μ*:*

$$\begin{aligned} \mathbf{i} \cdot \Pi^2(\mu) &= \Pi^2(\mu)(public) + \Pi^2(\mu)(private) + \Pi^2(\mu)(flag_1) + \Pi^2(\mu)(flag_2) \\ &= \mathbf{i} \cdot \Pi^2(\mu_0) = 1 \end{aligned}$$

This implies:

$$\Pi^2(\mu)(public) + \Pi^2(\mu)(private) \leq 1$$

So, the agent proves that it does not attempt to enter the private and the public place at the same time.

6 Conclusion

In this presentation we have introduced the formalism of elemntary object nets and its invariant calculus, which has the compositionality property, i.e. invariants of the whole system are deducible from the components. The usefulness of structural analysis combined with compositionality of multi-agent systems is obvious in the context of mobility, since the system is open and only parts of the systems are known in advance. Using this approach we could establish a correctness proof for our example scenario without any knowledge about the structure of the mobile agents in the system.

Bibliography

[BBPP04] M. A. Bednarczyk, L. Bernardinello, W. Pawlowski, L. Pomello. Modelling mobility with Petri hypernets. In Fiadeiro et al. (eds.), *Recent Trends in Algebraic Development Techniques (WADT 2004)*. Lecture Notes in Computer Science 3423, pp. 28–44. Springer-Verlag, 2004.

[Bus99] N. Busi. Mobile nets. In Ciancarini et al. (eds.), *Formal Methods for Open Object-Based Distributed Systems*. Volume 139, pp. 51–66. Kluwer, 1999.

[CGG00] L. Cardelli, A. D. Gordon, G. Ghelli. Ambient groups and mobility types. Technical report, Microsoft Research and University of Pisa, 2000.

[HEM05] K. Hoffmann, H. Ehrig, T. Mossakowski. High-Level Nets with Nets and Rules as Tokens. In *Application and Theory of Petri Nets and Other Models of Concurrency*. Lecture Notes in Computer Science 3536, pp. 268 – 288. Springer-Verlag, 2005.

[Hir02] K. Hiraishi. PN^2: An Elementary Model for Design and Analysis of Multi-agent Systems. In Arbab and Talcott (eds.), *Coordination Models and Languages, COORDINATION 2002*. Lecture Notes in Computer Science 2315, pp. 220–235. Springer-Verlag, 2002.

[KF06] M. Köhler, B. Farwer. Modelling Global and Local Name Spaces for Mobile Agents Using Object Nets. *Fundamenta Informaticae* 72(1-3):109–122, 2006.

[KF07] M. Köhler, B. Farwer. Object Nets for Mobility. In Kleijn and Yakovlev (eds.), *International Conference on Application and Theory of Petri Nets 2007*. Lecture Notes in Computer Science 4546, pp. 244–262. Springer-Verlag, 2007.

[KMR01] M. Köhler, D. Moldt, H. Rölke. Modeling the Behaviour of Petri Net Agents. In Colom and Koutny (eds.), *International Conference on Application and Theory of Petri Nets*. Lecture Notes in Computer Science 2075, pp. 224–241. Springer-Verlag, 2001.

[KMR03] M. Köhler, D. Moldt, H. Rölke. Modelling Mobility and Mobile Agents using Nets within Nets. In Aalst and Best (eds.), *International Conference on Application and Theory of Petri Nets 2003*. Lecture Notes in Computer Science 2679, pp. 121–140. Springer-Verlag, 2003.

[Köh06] M. Köhler. The Reachability Problem for Object Nets. In Moldt (ed.), *Proceedings of the Workshop on Modelling, object, components, and agents (MOCA'06)*. University of Hamburg, Department for Computer Science, 2006.

[Köh07] M. Köhler. Reachable Markings of Object Petri Nets. *Fundamenta Informaticae* 79(3-4):401 – 413, 2007.

[KB09] M. Köhler-Bußmeier. Hornets: Nets within Nets combined with Net Algebra. In Wolf and Franceschinis (eds.), *International Conference on Application and Theory of Petri Nets (ICATPN'2009)*. Lecture Notes in Computer Science. Springer-Verlag, 2009.

[KR03] M. Köhler, H. Rölke. Concurrency for Mobile Object-Net Systems. *Fundamenta Informaticae* 54(2-3), 2003.

[KR04] M. Köhler, H. Rölke. Properties of Object Petri Nets. In Cortadella and Reisig (eds.), *International Conference on Application and Theory of Petri Nets 2004*. Lecture Notes in Computer Science 3099, pp. 278–297. Springer-Verlag, 2004.

[Lak05] C. Lakos. A Petri Net View of Mobility. In *Formal Techniques for Networked and Distributed Systems (FORTE 2005)*. Lecture Notes in Computer Science 3731, pp. 174–188. Springer-Verlag, 2005.

[Lau87] K. Lautenbach. Linear algebraic techniques for place/transition nets. In Brauer et al. (eds.), *Petri Nets: Central Models and their Properties. Advances in Petri Nets 1986*. Lecture Notes in Computer Science 254, pp. 142–167. Springer-Verlag, 1987.

[Lom00] I. A. Lomazova. Nested Petri Nets – a Formalism for Specification of Multi-agent distributed systems. *Fundamenta Informaticae* 43(1-4):195–214, 2000.

[MPW92] R. Milner, J. Parrow, D. Walker. A calculus of mobile processes, parts 1-2. *Information and computation* 100(1):1–77, 1992.

[STC98] M. Silva, E. Teruel, J. M. Colom. Linear Algebraic and Linear Programming Techniques for the Analysis of Place/Transition Net Systems. In Reisig and Rozenberg (eds.), *Lecture Notes in Computer Science: Lectures on Petri Nets I: Basic Models*. Advances in Petri Nets 1491, pp. 309–373. Springer-Verlag, 1998.

[Val91] R. Valk. Modelling Concurrency by Task/Flow EN Systems. In *3rd Workshop on Concurrency and Compositionality*. GMD-Studien 191. Gesellschaft für Mathematik und Datenverarbeitung, St. Augustin, Bonn, 1991.

[Val98] R. Valk. Petri Nets as Token Objects: An Introduction to Elementary Object Nets. In Desel and Silva (eds.), *Application and Theory of Petri Nets*. Lecture Notes in Computer Science 1420, pp. 1–25. 1998.

[Val03] R. Valk. Object Petri nets: Using the nets-within-nets paradigm. In Desel et al. (eds.), *Advanced Course on Petri Nets 2003*. Lecture Notes in Computer Science 3098, pp. 819–848. Springer-Verlag, 2003.

[VC98] J. Vitek, G. Castagna. Seal: A Framework for Secure Mobile Computations. In *ICCL Workshop: Internet Programming Languages*. Pp. 47–77. 1998.

[XD00] D. Xu, Y. Deng. Modeling Mobile Agent Systems with High Level Petri Nets. In *IEEE International Conference on Systems, Man, and Cybernetics'2000*. 2000.