Petri Hypermats
for modeling mobile agents

Lucia Pomello

Università degli Studi di Milano - Bicocca
Dipartimento di Informatica, Sistemistica e Comunicazione

Moscow, 14/04/2014
Petri Hypernets for modeling mobile agents

Joint work with:

Marek Bednarczyk
Wiesław Pawłowski
Polish Academy of Science
IPI-PAN Gdańsk

Luca Bernardinello
Marco Mascheroni *
Università degli Studi
di Milano–Bicocca
Petri Hypernets for modeling mobile agents

Joint work with:

Marek Bednarczyk
Wiesław Pawłowski
Polish Academy of Science
IPI-PAN Gdańsk

Luca Bernardinello
Marco Mascheroni *
Universitá degli Studi
di Milano–Bicocca

* now at CERN
(Centre Européen Recherche Nucléaire)
Outline

1. Systems of mobile agents
2. Petri Hypernets, the model
3. From Petri Hypernets to a basic class of Petri nets
4. Generalized hypernets
5. Two case studies
6. Conclusions
Main concepts
- Agents are *autonomous* and *interacting*
- They are embedded into *locations*
- *Mobility* means an agent can change its location
- *Hierarchical* structure which may change
Systems of Mobile Agents

Main concepts

- Agents are *autonomous* and *interacting*
- They are embedded into *locations*
- *Mobility* means an agent can change its location
- *Hierarchical* structure which may change

Main aim

- A *formal* model based on *Petri net theory* for systems of mobile agents
The Nets-within-nets Paradigm

- Petri Nets $\Rightarrow$ Nets-within-nets
- Tokens of a Petri net can be nets themselves

Modeling Mobile Agents

- Agents modeled as *nets*, which are *tokens* inside other agents
- Environment is a special agent
- Hierarchy of nets/agents

---

1 R. Valk, Hamburg Univ., 1996
The Nets-within-nets Paradigm

- Petri Nets $\Rightarrow$ Nets-within-nets
- Tokens of a Petri net can be nets themselves

Modeling Mobile Agents

- Agents modeled as *nets*, which are *tokens* inside other agents
- Environment is a special agent
- Hierarchy of nets/agents

---

1. R. Valk, Hamburg Univ., 1996
The Nets-within-nets Paradigm and Mobile Agents

The Nets-within-nets Paradigm

- Petri Nets $\rightarrow$ Nets-within-nets
- Tokens of a Petri net can be nets themselves

Modeling Mobile Agents

- Agents modeled as nets, which are tokens inside other agents
- Environment is a special agent
- Hierarchy of nets/agents

---

$^1$R. Valk, Hamburg Univ., 1996
SECTION 2

Petri Hypernets, the model
Petri Hypernets – introduction

< \mathcal{N}, \mu >

- **Hypernet** \( \mathcal{N} \): a set of *agents*
- **Agent** \( A \): a *net*, synchronous product of state-machine nets (*modules*)
Petri Hypernets – introduction

\(< N, \mu >\)

- Hypernet \(N\): a set of agents
- Agent \(A\): a net, synchronous product of state-machine nets (modules)

synchronization: merging transitions

\[ N_1 || N_2 \]
Petri Hypernets – introduction

\(< \mathcal{N}, \mu >\)

- **Hypernet** \(\mathcal{N}\): a set of *agents*
- **Agent** \(A\): a *net*, synchronous product of state-machine nets (*modules*)
- Each agent is *located in a place of another agent*, and can move around (inside a state machine component)
Petri Hypernets – introduction

\[ \langle \mathcal{N}, \mu \rangle \]

- **Hypernet** \( \mathcal{N} \): a set of *agents*
- **Agent** \( A \): a *net*, synchronous product of state-machine nets (*modules*)
- Each agent is located *in a place of another agent*, and can move around (inside a state machine component)
- **Typing**
  - \( \sigma_1 : \text{Agents} \rightarrow \text{Sorts} \)
  - \( \sigma_2 : \text{Modules} \rightarrow \text{Sorts} \)
  - \( \sigma_2 : \Rightarrow \text{Places} \rightarrow \text{Sorts} \)
Petri Hypernets – introduction

\(< \mathcal{N}, \mu >\)

- **Hypernet** \(\mathcal{N}\): a set of *agents*
- **Agent** \(A\): a *net*, synchronous product of state-machine nets (*modules*)
- Each agent is located *in a place of another agent*, and can move around (inside a state machine component)
- **Typing**
  \(\sigma_1 : \text{Agents} \rightarrow \text{Sorts}\)
  \(\sigma_2 : \text{Modules} \rightarrow \text{Sorts} \quad (\quad \Rightarrow \quad \text{Places} \rightarrow \text{Sorts})\)
- **Hypermarking** \(\mu : \mathcal{N} \rightarrow \mathcal{P}_\mathcal{N}\) (*compatible* with the typing)
- *Tree-like* containment hierarchy.
The agent hierarchy may change: virtual places

*Virtual places* specify how tokens may change level in the hierarchy
The agent hierarchy may change: *virtual places*

*Virtual places* specify how tokens may change level in the hierarchy

1. Between an agent and its super-agent

---

![Diagram with Petri nets illustrating the change in level between an agent and its super-agent.](attachment:image)
Virtual places

2. Within the same agent: inter-module exchange of tokens

Lucia Pomello (DISCo)

Petri Hypernets

Moscow, 14/04/2014
2. Within the same agent: inter-module exchange of tokens
Example – modelling an airport

- An airport agent $A$ with modules for planes ($\pi$) and travellers ($\tau$)
Example – modelling an airport

- An airport agent $A$ with modules for planes ($\pi$) and travellers ($\tau$)
Example – modelling an airport

- An airport agent $A$ with modules for planes ($\pi$) and travellers ($\tau$)
Example – modelling an airport

- An airport agent $A$ with modules for planes ($\pi$) and travellers ($\tau$)
Dynamics—building a consortium

The state of the system (i.e., the hypermarking) changes by firing consortia

A *consortium* is made by:

- the name of a transition $t$
- a set of *active agents* (containing $t$) moving other agents
- a set of *passive agents* being moved by their *super-agents*

**Remark:** an agent can be both *active* and *passive*
Dynamics—building a consortium

The state of the system (i.e., the hypermarking) changes by firing consortia

A *consortium* is made by:

- the name of a transition \( t \)
- a set of *active agents* (containing \( t \)) moving other agents
- a set of *passive agents* being moved by their *super-agents*

**Remark:** an agent can be both *active* and *passive*

**Firing** a consortium

the \( t \) transitions are fired *synchronously*
Dynamics—firing a consortium

*board is enabled*
Dynamics—firing a consortium

board is enabled
Dynamics—firing a consortium

after the firing of *board*
Dynamics—firing a consortium

after the firing of board
Two independent consortia
Two independent consortia
Inter-level synchronization via short loop

\[ \gamma \]

\[ \alpha \]

\[ \beta \]

\[ \gamma \]

Lucia Pomello (DISCo)
boundedness

Petri Hypernets are such that there is *no* creation *no* destruction of agents
Remarks

boundedness

Petri Hypernets are such that there is no creation no destruction of agents

Theorem: Tree-like Structure Preservation

Let $\langle N, \mu \rangle$ be a Petri Hypernet, $\mathcal{M}$ a hypermarking, $\Gamma$ a consortium enabled in $\mathcal{M}$, and $\mathcal{M}'$ be the hypermarking reached after executing $\Gamma$. If the agent’s hierarchy induced by $\mathcal{M}$ is a tree, then the agent’s hierarchy induced by $\mathcal{M}'$ is also a tree.
SECTION 3

From Petri Hypernets to a basic Petri net class (1-safe nets)
From Petri Hypernets to 1-safe nets

\[ \mathcal{F} : < \mathcal{N}, \mu > \longrightarrow N = (B, E, F, M_\mu) \]
From Petri Hypernets to 1-safe nets

\[ \mathcal{F} : <\mathcal{N}, \mu> \rightarrow N = (B, E, F, M\mu) \]

- **B conditions**: \(<p, A>\) "agent A is in place p";
From Petri Hypernets to 1-safe nets

\[ \mathcal{F} : < \mathcal{N}, \mu > \rightarrow N = (B, E, F, M_\mu) \]

- **B conditions**: \(< p, A >\) "agent A is in place p";
- **E events**: "potential consortia";
From Petri Hypernets to 1-safe nets

\[ \mathcal{F} : < \mathcal{N}, \mu > \rightarrow N = (B, E, F, M_\mu) \]

- \(B\) conditions: \(< p, A >\) "agent A is in place p";
- \(E\) events: "potential consortia";
- \(< p, A > \in M_\mu \iff \mu(A) = p\).
From Petri Hypernets to 1-safe nets – an example

\[
\begin{align*}
\text{(deplane, ...)} & \quad (T1, c1) \\
\text{(P1, lg)} & \quad (T1, l) \\
\text{(T2, c1)} & \quad (deplane, ...)
\end{align*}
\]
From Petri Hypernets to 1-safe nets

\[ \mathcal{F} : < \mathcal{N}, \mu > \rightarrow N = (B, E, F, M_\mu) \]

\( N \) is such that:
- each event is balanced (i.e.: \(|pre(e)| = |post(e)|\));
- \( N \) is 1-safe;
From Petri Hypernets to 1-safe nets

\[ \mathcal{F} : < \mathcal{N}, \mu > \rightarrow N = (B, E, F, M_\mu) \]

\(N\) is such that:

- each event is balanced (i.e.: \(|\text{pre}(e)| = |\text{post}(e)|\));
- \(N\) is 1-safe;
- \(N\) is state machine decomposable;
From Petri Hypernets to 1-safe nets

\[ \mathcal{F} : < \mathcal{N}, \mu > \rightarrow N = (B, E, F, M_\mu) \]

- Each event is balanced (i.e., \( |pre(e)| = |post(e)| \));
- \( N \) is 1-safe;
- \( N \) is state machine decomposable;
- The hyper-marking graph of \( < \mathcal{N}, \mu > \) is isomorphic to the case graph of \( N \).
Hypermarkings of \( \langle \mathcal{N}, \mu \rangle \) and reachable markings of \( \mathcal{N} \)

**Theorem**

A consortium \( \Gamma \) is enabled at a hypermarking \( M \) of \( \langle \mathcal{N}, \mu \rangle \) iff the corresponding transition \( t_\Gamma \) is enabled at the marking \( f(M) \) of \( \mathcal{N} \).

The execution of \( t_\Gamma \) leads to a marking which corresponds to the hypermarking reached after executing \( \Gamma \).
Hypermarkings of $<\mathcal{N}, \mu>$ and reachable markings of $\mathcal{N}$

**Theorem**

A consortium $\Gamma$ is enabled at a hypermarking $M$ of $<\mathcal{N}, \mu>$ iff the corresponding transition $t_{\Gamma}$ is enabled at the marking $f(M)$ of $\mathcal{N}$.

The execution of $t_{\Gamma}$ leads to a marking which corresponds to the hypermarking reached after executing $\Gamma$.

**Consequences**

- All the analysis techniques available for 1-safe nets can be applied to Petri Hypernets;
- All the properties decidable for 1-safe nets are decidable for Hypernets, too.
Analysis of a Hypernet through the analysis of the 1-safe net

- place-invariants:
  - some specific ones identify the structural components of the Hypernet
  (for ex.: all places where an agent can be hosted)
From Petri Hypernets to 1-safe nets

Analysis of a Hypernet through the analysis of the 1-safe net

- place-invariants:
  - some specific ones identify the structural components of the Hypernet
    (for ex.: all places where an agent can be hosted)
  - others are not so easily identifiable
    (for ex.: they contain places from different modules / agents)

...
From Petri Hypernets to 1-safe nets

Analysis of a Hyernet through the analysis of the 1-safe net

- place-invariants:
  - some specific ones identify the structural components of the Hyernet
    (for ex.: all places where an agent can be hosted)
  - others are not so easily identifiable
    (for ex.: they contain places from different modules / agents)

- ...

"In the future":
define invariants and other analysis technique directly on Hyernets.
Further work on Petri Hypernets

- Agent Aware Transition systems, as a behavioral model of mobile agents and as a basis for model checking Petri Hypernets. ²

- A logic for specifying temporal and structural properties of Petri Hypernets, towards model checking (by the IPI-PAN group). ³

---


SECTION 4

From Petri Hypernets to Generalized hypertnets
Motivations

Petri Hypernets have some constraints which can be too strict, e.g.:

- typing of agents,
- the modular structure of agents, ...
From Petri Hypernets to Generalized hypertnets

**Motivations**

Petri Hypernets have some constraints which can be too strict, e.g.:
- typing of agents,
- the modular structure of agents, ...

**New Features**

- Use of *paths* ⇒ weighted arcs
- Multi-typed agents
- A more flexible synchronization between adjacent agents

---

4 Marco Mascheroni’s PhD thesis, University of Milano -Bicocca, 2011
Generalized hypernet Consortia

The state of the system (i.e., the hypermarking) changes by firing consortia

Elements of a Consortium

- A set of transitions (with the same label is chosen)
Generalized hypernet Consortia

The state of the system (i.e., the hypermarking) changes by firing consortia

Elements of a Consortium

- A set of transitions (with the same label is chosen)
Generalized hypernet Consortia

The state of the system (i.e., the hypermarking) changes by firing consortia

Elements of a Consortium

- A set of transitions (with the same label is chosen)
- Binding virtual places
Generalized hypernet Consortia

The state of the system (i.e., the hypermarking) changes by firing consortia

Elements of a Consortium
- A set of transitions (with the same label is chosen)
- Binding virtual places
- Agent/Path association
Generalized hypernet Consortia

The state of the system (i.e., the hypermarking) changes by firing consortia

Elements of a Consortium

- A set of transitions (with the same label is chosen)
- Binding virtual places
- Agent/Path association
- The consortium can fire
Hypermarking graph of Generalized hypernets

A0
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>A1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A0</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A3</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A4</td>
</tr>
</tbody>
</table>

A1
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>A1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A0</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A3</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A4</td>
</tr>
</tbody>
</table>

A2
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>A2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A0</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A3</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A4</td>
</tr>
</tbody>
</table>

A0
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>A0</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A3</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>A4</td>
</tr>
</tbody>
</table>
Hypermarking graph of Generalized hypernets
Hypermarking graph of Generalized hypernets
Generalized Hypernets: further results

- From Generalized Hypernets to 1-safe nets
Generalized Hypernets: further results

- From Generalized Hypernets to 1-safe nets
- Definition of Unfolding of a Generalized Hypernet
SECTION 5

Two case studies
5.a) Modeling a Class of Membrane Systems

P Systems with Symport-Antiport Rules
- Bioinspired computational models
- Nested membranes which contain molecules and rules
- Rules change the configuration of the system
5.a) Modeling a Class of Membrane Systems

P Systems with Sympor-Antiport Rules
- Bioinspired computational models
- Nested membranes which contain molecules and rules
- Rules change the configuration of the system

Modeling P systems
- Each membrane is an agent
- Molecules are tokens
- Rules are transitions
- Bijection $g$ between hypermarkings and configurations
Theorem

Let $\Pi$ be a P system, and let $<\mathcal{N}, \mu>$ be the Hypernet which models it. If a set of rules $\rho$ is enabled in a configuration $C$ of $\Pi$, then the set of consortia $U$, which correspond to $\rho$, are enabled in the hypermarking $g(C)$. If $C'$ is the configuration reached after executing $\rho$, then $g(C')$ is the hypermarking reached after firing $U$. 

---

Theorem

Let $\Pi$ be a P system, and let $\langle \mathcal{N}, \mu \rangle$ be the Hypernet which models it. If a set of rules $\rho$ is enabled in a configuration $C$ of $\Pi$, then the set of consortia $U$, which correspond to $\rho$, are enabled in the hypermarking $g(C)$. If $C'$ is the configuration reached after executing $\rho$, then $g(C')$ is the hypermarking reached after firing $U$.

Possible Applications

Then it is possible to use Petri nets analysis techniques to analyze P system.

---

5.b) Modeling Grid Applications

Grid Computing

- Analysis of data which are distributed in different data centers at different levels
- Final user should not care of data distribution
- Tools, which make easier the access to the Grid, are developed

---

5.b) Modeling Grid Applications

- Is the Nets Within Nets Paradigm suitable to model/specify applications which use the Grid technology?

- Answer by modeling a Grid distributed data analysis tool serving the community of the Compact Muon Solenoid (CMS) experiment at the CERN Large Hadron Collider (LHC).
The CRAB Tool

- Split user’s work into jobs and analyze datasets in parallel through the Grid
- CRAB (CMS Remote Analysis Builder) has a client-server architecture
- Sends/monitors/retrieves output of jobs from the Grid
Modeling Grid Applications

The submission use case of the CMS Remote Analysis Builder (CRAB) have been modeled by using Nets-Within-Nets, and in particular by using the tool RENEW (developed in Hamburg Univ.)

Results

Nets within nets seems to be a promising approach for modeling Grid applications:
- The hierarchical structure of the system is represented in the model;
- some bugs in the software have been identified;
Modeling Grid Applications

The submission use case of the CMS Remote Analysis Builder (CRAB) have been modeled by using Nets-Within-Nets, and in particular by using the tool RENEW (developed in Hamburg Univ.).

Results

Nets within nets seems to be a promising approach for modeling Grid applications:

- The hierarchical structure of the system is represented in the model;
- some bugs in the software have been identified;
- Marco Mascheroni got a job at CERN, ::))), he is now in the group developing a new version of the tool (CRAB3).
SECTION 6

Conclusions
Conclusions

Petri Hypernets: a summary

A hierarchical model for mobile agents in the *nets-within-nets* paradigm

- A Petri Hypernet is a collection of *modular* nets
  (obtained by composing *state machines* through transition synchronizations)
Conclusions

**Petri Hypernets: a summary**

A hierarchical model for mobile agents in the *nets-within-nets* paradigm

- A *Petri Hypernet* is a collection of *modular* nets (obtained by composing *state machines* through transition synchronizations)

- A *Hypermarking* defines a *hierarchy* on the nets

- The hierarchy *dynamically changes*
## Petri Hypernets: a summary

A hierarchical model for mobile agents in the *nets-within-nets* paradigm

- A **Petri Hypernet** is a collection of *modular* nets (obtained by composing *state machines* through transition synchronizations)

- A **Hypermarking** defines a *hierarchy* on the nets

- The hierarchy *dynamically changes*

- Its semantics is given in terms of a basic net class
Petri Hypernets: a summary

A hierarchical model for mobile agents in the *nets-within-nets* paradigm

- A Petri Hypernet is a collection of *modular* nets (obtained by composing *state machines* through transition synchronizations)

- A Hypermarking defines a *hierarchy* on the nets

- The hierarchy *dynamically changes*

- Its semantics is given in terms of a basic net class

- They have been generalized and applied
Future work . . .

- relax some constraints of the model, e.g.:
  - add the possibility to create/destroy agents;
  - allow interactions among agents in the same location without the participation of their superagent

---


Future work . . .

- relax some constraints of the model, e.g.:
  - add the possibility to create/destroy agents;
  - allow interactions among agents in the same location without the participation of their superagent

- develop techniques for property analysis, e.g.: Linear invariants, . . . ;

---


Future work . . .

- relax some constraints of the model, e.g.:
  - add the possibility to create/destroy agents;
  - allow interactions among agents in the same location without the participation of their superagent

- develop techniques for property analysis, e.g.: Linear invariants, . . .

- define synchronous product of Petri (Genralized) hypernets;

---


Future work . . .

- relax some constraints of the model, e.g.:
  - add the possibility to create/destroy agents;
  - allow interactions among agents in the same location without the participation of their superagent

- develop techniques for property analysis, e.g.: Linear invariants, . . . ;

- define synchronous product of Petri ( / Genralized) hypernets;

- develop a software tool, on the basis of already developed prototypes:
  - an *editor-simulator* of Hypermets with a model checker ⁷ (in Gdansk)
  - a plugin for Renew based on Hypermets (in collaboration with Hamburg Univ.). ⁸

---


Future work . . .

- relax some constraints of the model, e.g.:
  - add the possibility to create/destroy agents;
  - allow interactions among agents in the same location without the participation of their superagent
- develop techniques for property analysis, e.g.: Linear invariants, . . . ;
- define synchronous product of Petri (/ Generalized) hypernets;
- develop a software tool, on the basis of already developed prototypes:
  - an *editor-simulator* of Hypernets with a model checker \(^7\) (in Gdansk)
  - a plugin for Renew based on Hypernets (in collaboration with Hamburg Univ.). \(^8\)

- a possible collaboration with the PAIS Lab combining *Nested Petri Nets* and *Peri Hypernets*?

---


References

THANK YOU for your attention!

Спасибо большое!

GRAZIE!