Hierarchical Design Method
for Multi-Agent Systems

Bouneb Messaouda, University of El Arbi Ben M’hidi, Oum El Bouaghi, Algeria

Saïdouni Djamel Eddine, Department of Fundamental Informatics and Its Applications,
University Constantine 2-Abedlhamid Mehri, Constantine, Algeria

ABSTRACT
This paper proposes a new hierarchical design method for the specification and the verification of multi agent systems (MAS). For this purpose, the authors propose the model of Refinable Recursive Petri Nets (RRPN) under a maximality semantics. In this model, a notion of undefined transitions is considered. The underlying semantics model is the Abstract Maximality-based Labeled Transition System (AMLTS). Hence, the model supports a definition of a hierarchical design methodology. The example of goods transportation is used for illustrating the approach. For the system assessment, the properties are expressed in CTL logic and verified using the verification environment FOCOVE (Formal Concurrency Verification Environment).

Keywords: Action Refinement, Hierarchical Design, Maximality-Based Labeled Transition System, Multi-Agent Systems, Recursive Petri Nets

INTRODUCTION
The problem of designing multi-agent systems (MAS) is at the heart of many researches. This is an important issue that refers to a crucial question: how to link a global description of a task and agents whose behavior depends on a view that is necessarily partial and local in the same task.


The process of designing multi agent systems involves the use of a design methodology based on a specification model. In fact, in order to design a multi agent systems, many formalisms have been proposed such as Z-language (Regayeg, Kacem & Jmaiel, 2005), Maude (Mokhati, F., Boudiaf, N., Badri, M., & Badri, L. 2007), Logic (Lomuscio & Sergot, 2003), Recursive Petri Nets (RPN for short) (Seghrrouchni & Haddad, 1996), Synchronized Petri Nets (SyPN

DOI: 10.4018/IJATS.2015040104
for short) (Kouah, Saïdouni & Ilié, 2013) (Kouah & Saïdouni, 2014). The common feature of these models is that they are based on the interleaving semantics which considers that actions are atomic and with null duration. This semantics interprets the parallel execution of actions by their interleaved ones. Consequently, the interleaving semantics is not appropriate for design approaches based on action refinement.


In this context the maximality semantics, through the model of the Maximality-based Labeled Transition Systems (MLTS for short), was used for the characterization of concurrent systems. For this purpose, in (Saïdouni & Bounub, 2013) a maximality-based operational semantics has been proposed for recursive Petri net model.

In this paper, we offer a hierarchical design method that overcomes the complexity of designing multi agent systems. To attain this goal we extend the model of recursive Petri net by considering refinable transitions. The model will be named Refinable Recursive Petri net. The proposed model is based on the maximality semantics. As for recursive Petri net model, dynamic behaviors are considered through abstract transitions.

These abstract transitions can be used for designing a system following a top down manner. Hence, at some design level, the behavior details of a transition may be abstracted. They will be exhibited further more in next refined specifications. In the first step, abstract transitions behaviors are undefined. These behaviors are gradually introduced along the design trajectory. In this manner, system components are integrated gradually; the initial specification is then the most abstract one. This remark leads us to label abstract transitions, whose behaviors are undefined by the symbol $\bot$ ("bottom").

As an example, let us consider systems of Figure 1 where transitions $t_1$ and $t_4$ are abstract transitions, at this level, the difference between the systems may not be seen. The labeled transition system associated with the two Petri nets is given by Figures 2 and 3.

This example shows that the difference between two undefined transitions may be seen after their refinement.

**Figure 1. Recursive Petri net**

![Recursive Petri net](image-url)
Multiagent Learning on Traffic Lights Control: Effects of Using Shared Information
www.igi-global.com/chapter/multiagent-learning-traffic-lights-control/26945?camid=4v1a

Autonomous Specialization in a Multi-Robot System using Evolving Neural Networks
www.igi-global.com/chapter/autonomous-specialization-multi-robot-system/46204?camid=4v1a