# Modeling Mobility Agents in Supervisory and Controlling Systems ased on Nets within Nets (ICCAS2005)

Hu iaohui*,**, Dang Jianwu *, Zhou ingshe**
* Key Laboratory of Opto-Electronic Technology and Intelligent Control (Lanzhou Jiaotong University), Ministry of Education, Lanzhou Jiaotong University, Lanzhou, China
(Tel:+86-0931-4956029;hxh 6302@163.com)
** Department of Computer Science and Engineering , Northwestern Polytechnical University , i an , China
(Tel:+86-029-8849289;zhouxsh@wnpu.mail.cn)

**Abstract**: The goal of our research is to develop a formal modeling methodology for supervisory and controlling systems that have artificially intelligent features. This approach is agent-based and central to the development of the model of mobility agent considering reactivity for real-time purpose and deliberation for optimal realization and safe-fail problems for critical systems like Intelligent Transportation Systems by high-level Petri net. By using nets within nets we investigate the concurrency of the system and the agent in one model without losing the needed abstraction, and synchronous channels are introduced to denote the coordination and communication. Finally an example is demonstrated.

**Keywords:** Agent, Petri nets, Model, MAS, Communication.

## 1. Introduction

In supervisory and controlling systems like Intelligent Transportation Systems (ITS), mobility is an important concept for modeling and verification. We treat mobility entities in the systems as agents that integrate and connect together to form MAS. The main problem is how to model mobility agent in a suitable manner without losing formal accuracy. The modeling language should have the features of graphical representation with verification ability by using a formal semantics. In the paper we use the concept of nets within nets to describe mobility agent, its coordination and communication mechanisms in MAS. It is known that one of the technical impediments to the widespread adoption of MAS technologies is lacking of a systematic methodology to synthesize MAS because many architectures are ad hoc [1]. The high level Petri nets-based models provide valuable methods to MAS research in general.

The remainder of the paper is organized as follows. In section 2, the basic concepts of nets within nets are given; in section 3, we demonstrate the extended BDI architecture; in section 4, we describe how to use nets within nets to specify supervisory and controlling systems as MAS, and the coordination and communication mechanisms of the system; finally some conclusions are drawn.

## 2 asic concepts of nets within nets

Petri net is assembled from places and transitions. Places represent resources that can be available or not, or conditions that may be fulfilled. Transitions are the active part of a net. A transition removes resources or conditions from places and inserts them into other places. This is determined by arcs, which are directed from places to transitions and from transitions to places [2].

Nets within nets are high-level Petri nets that are suited for the specification and modeling of complex distributed systems like supervisory and controlling systems in hierarchical way [3]. Nets within nets offer some extensions related to colored Petri nets: nets as token objects representing mobility agent, communication via synchronous channels. The overall system net specifies MAS structure and locations of the systems, and agent nets as tokens can show different agent architectures, for example, the extended BDI agent architecture.

There are net instances in nets within nets. Net instances are similar to objects, and different instances of the same net can take different states at the same time and are independent from each other in all respects. Certainly hierarchies of net within net relationships are permitted, and a system net containing agent net tokens may itself be an agent net token of another net. A synchronous channel permits a fusion of transitions (two at a time) for the duration of one occurrence. Channels are directed, i.e. exactly one of the two fused transitions indicates the net instance in which the counterpart of the channel is located. The other transition can correspondingly be addressed from any net instance. The transferred information via a synchronous channel can take place bi-directional and is also possible within one net instance. It is possible to synchronize more than two transitions at a time by inscribing one transition with several synchronous channels. In addition to the usual arc types, there can be different types of arcs in nets within nets for the rich meaning.

## 3.Agent

Here, the agent nets have the similar structure implemented in extended BDI [4]. The modified agent model has functional parts of beliefs, desires, intentions, commitments and capabilities to reflect the agent s mental states, which decide the conditions of taking agent s behaviors. Meanwhile the model considers reactivity for real-time purpose and deliberation for optimal realization and safe-fail problems for critical systems like ITS.

In extended BDI architecture, the agent model represented by Petri nets concentrates on the agent s behaviors, which are composed of a set of behavioral rules and is described by a set of interactions and agent s intentions. Agent s intentions are also expressed by Petri nets, which are written with some regulations. There are several types of behavioral rules. The deliberative behavioral rules have complex decision making processing, and take sequence of actions to achieve the optimal plan. The reactive behavioral rules are responsible for real-time decision making, getting plan and taking actions quickly. In this case, agent takes time as little as possible to have simple decision-making, take critical actions according to the related plans and meet deadlines. For the critical systems like ITS, the special kind of behavioral rules called safe-fail behavioral rules are also set up for the reason of system s safe operations in case of abnormal conditions.

Fig.1 shows the extended BDI agent model with three types of behavioral rules. Transition Evaluation decides on what control strategy to take according to recent mental states and input messages of mobility agent provided by the place Knowledge Base and place Input Messages. The outputs of transition Evaluation are connected with transitions DM1, DM2, DM3, which are responsible for selecting valid intentions from set of deliberative plans, real-time plans and fail-safe plans respectively. With the different intentions, the transitions BR1, BR2, BR3 are for behavioral rules and take

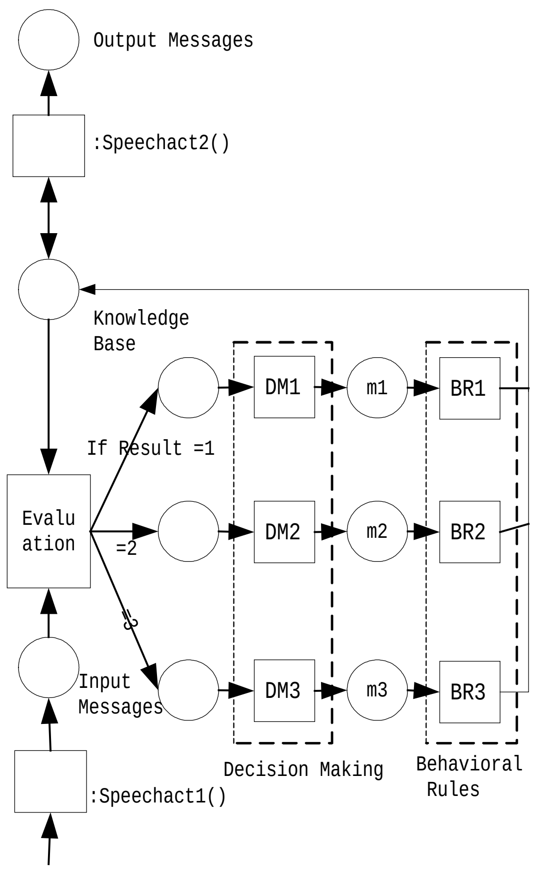



Fig.1 Extended BDI agent model

different suitable actions for changing scenarios. In the mobility agent model, the synchronous channels of nets within nets are used for representing the communication between mobility agent and system net. The uplinks, :SpeechAct1() and ;SpeechAct2() which are related with the performatives of KQML[5] or FIPA[ 6] according to speech act theory, describe how to express input and output of the messages of the mobility agent.

The transitions Evaluation, DM1, DM2, DM3, BR1, BR2 and BR3 can further be hierarchically decomposed into detailed Petri nets, and Fig.2 is the typical structure of behavioral rule transitions used in ITS. With the decided intention (place m1), related beliefs (place b1) and desires (place d1), transition c1 representing commitments will take series of actions including moving mobility agent ahead in certain modes and change the agent mental state to (b2, d2). After that, transition J judges whether mobility agent will be in the recent district or not. If mobility agent is in the recent district, the position information of agent would be like this: Agent.District = 1 and Agent.RunDistance = 2000. If not, Agent.District = 2 and Agent.RunDistance = 50, and in this

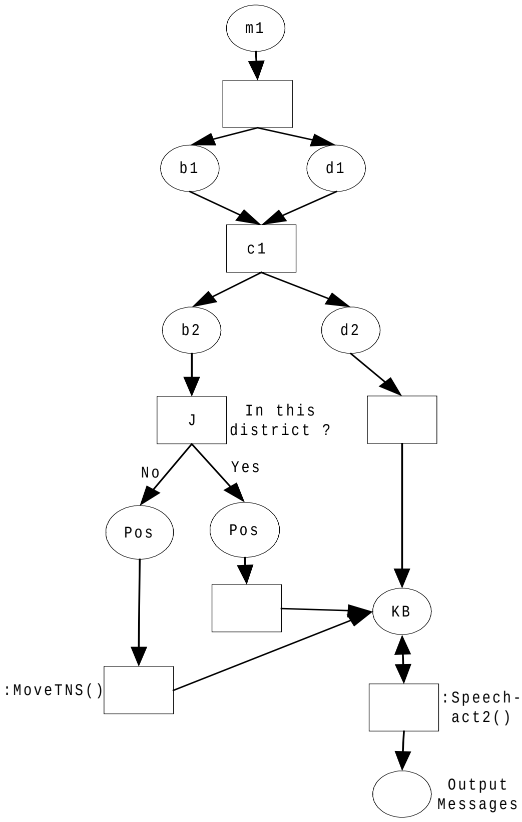


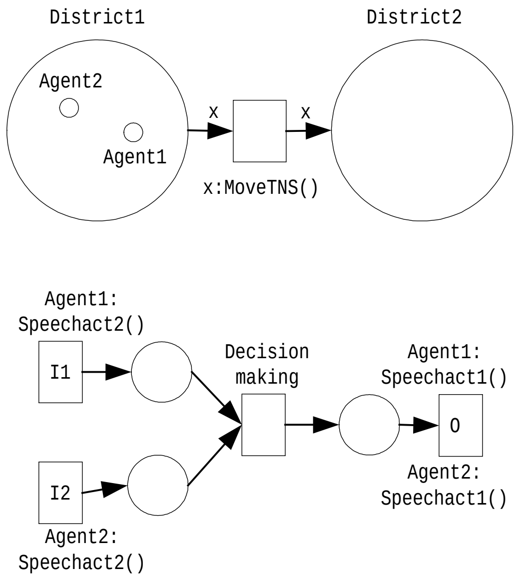

Fig.2 Behavioral rules module of Agent

case, the uplink :MoveTNS() in this Fig.2 and downlink x:MoveTNS() in Fig.3 will work together to show agent moving from one district to next district in system net of nets within nets.

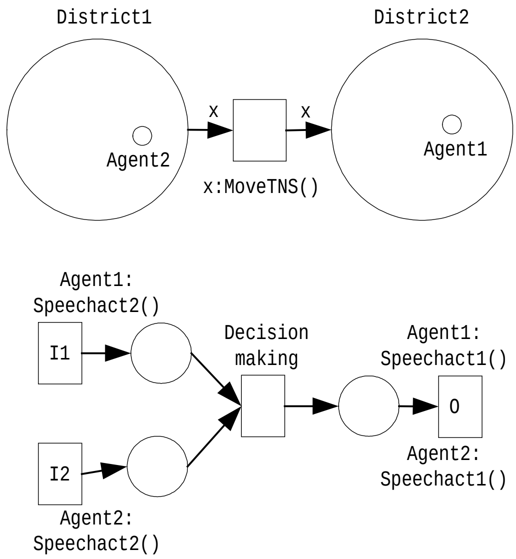
## 4. Modeling MAS architecture with mobility entities by using nets within nets

### 4.1 Modeling MAS architecture

The Supervisory and Controlling Systems like ITS can be treated as MAS, in which there are static entities and mobility entities viewed as agents. The overall system is divided into separate locations. An important point of mobility is the embeddedness of the mobile entity, which means that each entity is embedded in a local environment that assists the entity by offering some services and restricts it by declining others. The systems we are considering usually show several differences between the locations and hence are more interesting to model, since these differences cause the complexity of the systems.




(a)




(b)

Fig.3 an example of MAS architecture with mobility agents

To give an example of ITS, we consider the system with a two-level hierarchy represented by nets within nets. The overall system and its environment are presented by system net as first level hierarchy. In the system net, the layout of transportation system is drawn as Petri nets and supervisory equipment is modeled as supervisory agent that is responsible for the coordination of the whole transportation system. The net token is called agent net that is the second level hierarchy and is represented with the architecture of mobility agent shown in Fig.1.

A simple example of road transportation system in a city is shown as Fig. 3. The places District1 and District2 represent different districts in the city. Originally, there are two cars denoted as Agent1 and Agent2 running in the District1 as shown in Fig.3 (a), and we suppose that Agent1 is ahead of Agent2. Under certain circumstances, Agent1 will leave District1 and enter into District2, which can be represented by synchronous channel. In Fig.3 (a), Agent1 first can be bound to the arc variable x when Agent1 wants to across the border. Then by using downlink x:MoveTNS() of the system net and uplink :MoveTNS() of Agent1 as a token net, the concept of crossing border is clearly represented. Fig.3 (b) shows the situation in which Agent1 enters into District2 and Agent2 is still in District1.

The supervisory agent is located in the system net. Through downlinks Agent1:Speechact2() and Agent2:Speechact2() of system net and uplinks :Speechact2() of Agent1 and Agent2, we can imitate message passing from Agent1 and Agent2 to supervisory agent. After getting the information, transition Decisionmaking will dispose it, get optimal transportation plan and send to Agent1 and Agent2 through channels Speechact1().

The example gives an idea how the interplay between mobility agent net and system net can be used to model mobility entities moving through a system net, where the system net offers or denies possibilities to move around while the mobility agent net moves at the right time by activating a respective transition that is inscribed with the counterpart of the channel of the transition of the system net. Without the viewpoint of agent nets as tokens, we would have to encode the agent somehow, for example as a data structure. The disadvantage of such an approach is that the inner actions of the mobile entity cannot be modeled directly, so, they have to be lifted up 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 losing the needed abstraction.

### 4.2 The coordination and communication

The supervisory and controlling systems are complicated systems that need many agents in the system to coordinate and achieve the general and local goals. There are several types of coordination represented by nets within nets by using synchronous channels. The communication delay and degradation should be considered when modeling coordination mechanism.

In general, multiple levels of synchronization are suspect from a methodical point of view, because they tend to be difficult to understand. Petri nets excel at displaying control flows and it seems that synchronous channels should not be used to encapsulate complex control flows or even loops. It is best to use channels where they show their greatest potential, namely synchronization, communication, and atomic modifications.

## 5. Conclusion

In supervisory and controlling systems like ITS, mobility is an important concept for modeling and verification. In the paper we use nets within nets to model mobility agent and form MAS. The communication of the distributed system is expressed by synchronous channel in clear and natural ways.

## REFERENCES

[1] N. R. Jennings, K. Sycara and M. Wooldridge, A Roadmap of Agent Research and Development, Autonomous Agents and Multi-Agent Systems, vol. 1, pp. 275-306, 1998.

[2] Murata, Petri Nets: Properties, Analysis and Applications, Proc. of the IEEE, Vol. 77, No. 4, pp. 541-580, 1989.

[3] Michael Kohler, Daniel Moldt, and Heiko Rolke, Modelling the structure and behaviour of Petri net agents, Proc. of the 22nd Conference on Application and Theory of Petri Nets, Vol. 2075, pp. 224－241, 2001.

[4] [Kinny et al. 1996] D. Kinny, M. Georgeff, and A. Rao, A Methodology and Modeling Technique for Systems of BDI Agents, Proc. of the Seventh European Workshop on Modeling Autonomous Agents in a Multi-Agent World, Lecture Notes in Artificial Intelligence, Vol. 1038, pp. 56-71, 1996,

[5] Tim Finin, annis Labrou, and James Mayfield, KQML as an agent communication language, In Jeff Bradshaw (ed.), *Software Agents*, MIT Press, Cambridge, 1997.

[6] F. Bellifemine, A. Poggi, G. Rimassa, JADE - A FIPA-compliant Agent Framework, Proc. of 4th International Conference on the Practical Application of Intelligent Agent and Multi Agent Technology, pp. 97-108, 1999.

### Acknowledgments

This research is supported by Nature and Science Foundation of Gansu, China, Grant NO. 3ZS042-B25-025, Qing-Lan Project of Lanzhou Jiaotong University, Gansu, China and the Opening Foundation of the Key Laboratory of Opto-Electronic Technology and Intelligent Control (Lanzhou Jiaotong University), Ministry of Education, China, Grant NO. K040109.