A Goal Based Approach on top of Petri Nets

Nejm Saadallah and Benoit Daireaux

IRIS 4068 Stavanger Norway nejm.saadallah@iris.no , IRIS 4068 Stavanger Norway benoit.daireaux@iris.no

Abstract. This poster presents ongoing work and mainly proposes a way to model goals on a Petri net model. We consider that the basic functioning of many machines can be captured in a Petri net model, while the environment where the machine is deployed is often too complex to be modelled in Petri nets. We handle the influence of the environment on the choice of operations by the so-called external agents, and show how these agent's goals could be studied before real deployment.

Keywords: *Control System, Agents, Supervisory Control, Petri Net, Model building*

1 Problem Formulation

The basic idea of our approach is to use Petri nets [6] to model the dynamic of machines [1, 3, 5, 4], regardless of the environments in which they are deployed, and use the notion of agents'goals to model the influences of the environments on the machine's behaviour. Our approach is motivated by two facts. The first fact is that basic functioning of many machines can be captured in a Petri net model. The second fact is that the environments in which a given machine is deployed are often hard to model within Petri nets. In contrast with synchronous Petri nets [2] which include sensory data into transition firing rules, we consider sensory data as inputs to external agents that need to interact with the machine.

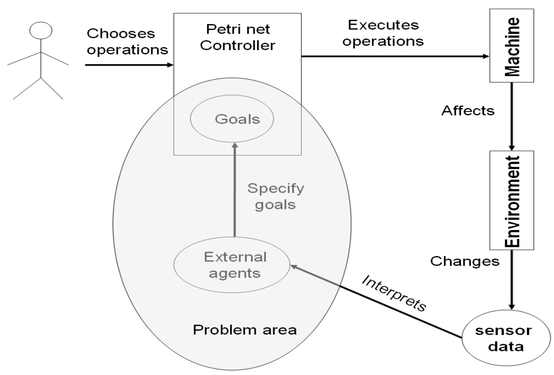

Fig. 1. Actions are chosen according to a logic control that is captured using the concept of control interpreted Petri nets. The machine commands affect the surrounding environment, which is reflected in sensory data. An external agent could be a software or human agent continuously analyses sensory data, and triggers actions.

In this work we are interested in finding a systematic approach for studying the behaviour of agents, based on their respective goals. Given a set of agents acting on a machine that is modelled in Petri net, how can we classify the behaviour of these agents? To the system designer, this approach is intended to give an understanding of the system prior to its implementation. Modelling the execution policy, that is, how the interaction between the agents and the Petri net model could be implemented is not addressed in this paper, but is shortly discussed in Section 3.

2 Modelling Goals

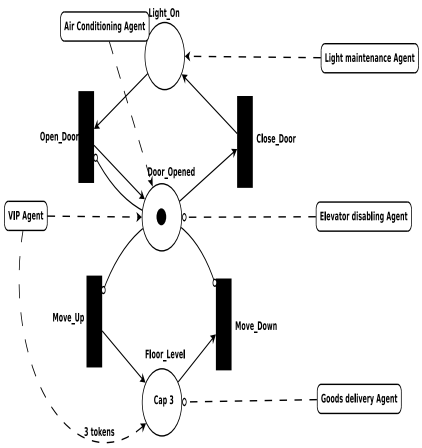

Fig. 2. Shows a simple elevator enhanced with 5 agents. **Light maintenance agent** will trigger every period of time to check whether the light lamp is still working or not. **Elevator disabling agent** will trigger if there is a fire in the building and no person in the elevator, to keep the door closed. **Goods delivery agent** moves the elevator to the floor 0. **VIP agent** requests the elevator to be at the third floor with the door open. **Air conditioning agent**, when triggered the door of the elevator stays open for a period of time.

The five agents shown in Figure 2, are categorised according to four relations as illustrated in Table 1. The four relations are defined as follows:

Distinctly Inclusive goals: We say that a goal g_a distinctly includes g_b if by achieving g_a, g_b is also achieved.

Mutually Inclusive goals: We say that two goals g_a and g_b are mutually inclusive if g_a distinctly includes g_b, and g_b distinctly includes g_a.

Partially Inclusive goals: We say that g_a partially include g_b, when only some markings that achieve g_a also achieve g_b but not all of them.

Mutually Exclusive goals: We say that two goals g_a and g_b are mutually exclusive when they can not be achieved together.

3 Conclusion and Future Work

In this paper we have addressed some aspects that are related to machines deployed in complex environments. We believe that the basic functioning of many

Table 1. Summary of the relations between the five goals of Figure 2

Goals	Mutual Exclusion	Mutual Inclusion	Distinct Inclusion	Partial Inclusion
Light maintenance g_1	g_4, g_5	g_2	g_2	g_3 via M_1
Elevator disabling g_2	g_4, g_5	g_1	g_1	g_3 via M_1
Goods delivery g_3	g_4	No	No	g_5 via M_0, g_1 and g_2 via M_1
VIP g_4	g_1,g_2,g_3	No	g_5	No
Air conditioning g_5	g_1,g_2	No	No	g_3 via M_0, g_4 via M_7

machines can be captured in a Petri net model, while the environments where the machines are deployed are often too complex to model in Petri nets. We consider that the influence of the environment on the machine is handled by agents, and raise the the following question: how agents acting on a machine can be categorised? We answer the question by introducing four properties, and use these properties to analyse some behavioural aspects, prior to system implementation as illustrated in Table 1.

We focused on the off-line analyses of agents acting on a machine, but we have not studied the on-line problematic, in other words the execution policy. When several agents are involved, which one should have the priority to execute? Another interesting question would be to find the set of necessary agents that guaranty some safety levels. To be more precise, could we provide a Petri net modelled machine with a set of safety agents, such that even in the presence of other faulty agents, the system guaranties the specified safety level? We believe that this work could be done within the Petri net formalism, and will be the subject of our future efforts.

References

1. Christos G. Cassandras and Stephane Lafortune. Introduction to discrete event systems. 2006.
2. René David and Hassane Alla. *Discrete, Continuous, and Hybrid Petri Nets.* Springer, 1 edition, November 2004.
3. B. Hrz and M. C. Zhou. *Modeling and Control of Discrete-event Dynamic Systems: with Petri Nets and Other Tools.* Springer Publishing Company, Incorporated, 2nd edition, 2007.
4. Vedran Kordic, editor. *Petri Net, Theory and Applications.* I-Tech Education and Publishing, 2007.
5. Tadao Murata. Petri nets: Properties, analysis and applications. pages 541–580, April 1989. NewsletterInfo: 33Published as Proceedings of the IEEE, volume 77, number 4.
6. Carl Adam Petri. *Communication with automata.* PhD thesis, Univ. Hamburg, 1966.