

# A REFERENCE MODEL FOR DISTRIBUTED DECISION MAKING THROUGH A MULTI-AGENT APPROACH

---

Ilaria Baffo<sup>1,2</sup>, Giuseppe Confessore<sup>1</sup>,  
Giacomo Liotta<sup>1</sup>, Giuseppe Stecca<sup>1</sup>

<sup>1</sup> Istituto di Tecnologie Industriali e Automazione  
Consiglio Nazionale delle Ricerche

<sup>2</sup> Dipartimento di Ingegneria dell'Impresa  
Università di Roma "Tor Vergata", ITALY

{ilaria.baffo, g.confessore, g.liotta, g.stecca}@itia.cnr.it

*Multi-agent modeling is widely used in collaborative networks for the ability to represent complexity derived from interrelationship. Nevertheless we noticed in the literature a lack of guidelines in formal modeling. In this paper we analyze a reference model to develop multi-agent models able to be applied to a variety of fields. The presented method brings together operations research techniques and UML notation. The reference model is validated in different application fields, in particular in this paper an application to a shoe manufacturing plant is presented.*

## 1. INTRODUCTION

During last years, a lot of studies done in field of complex system management have shown how modeling is one of the key activities in understanding, designing, implementing, and operating systems (Camarinha-Matos *et al*, 2006). A model lets to predict system behavior, evaluate different system problem solutions, test different alternatives and analyze their effect without modifying the real system. The huge potentials offered by models can be exploited particularly in the representation of complex systems such as collaborative networks. Camarinha-Matos and Afsarmanesh, in (Camarinha-Matos *et al*, 2005), defined a collaborative network as a network consisting of autonomous, distributed, heterogeneous entities that collaborate to better achieve common or compatible goals. Following this definition, the application to Multi Agent System (MAS) to collaborative networks modeling results natural. The MAS are widely studied in literature as a method able to represent dynamic and distributed system with several decision makers having different information domains. We can often observe some lacks in investigation of *the problem solving ability of intelligent agents in a multi-agent setting* and a variety of representation methods. In this paper we present a reference model that integrates two types of approach for the models creation in the MAS field. The main purpose is

to provide a set of directives able to support the designer in the system modeling phase of MAS development while indicating in which scenario is more convenient to adopt an approach over another one. The remainder of this paper is structured as follows. Section 2 presents a reference model with a short discussion about model presented in the literature for MAS building. In section 3 an application to flexible manufacturing is used to show like using the guideline offered by the Reference Model for MAS creation. In section 4 two more applications are briefly presented in order to validate the reference model proposed. Finally, conclusions and future works are discussed in section 5.

## 2. REFERENCE MODEL

### 2.1 Methods for multi - agent model development

The concept of Multi-Agent System is widely studied in computer science field (Jennings *et al*, 1995). Ever since several researchers studied the MAS theory, offering a lot of agent definitions and application fields of this theory. The different models presented in the literature can be classified into two research traditions: (i) the first one IT-oriented and close to MAS origin, (ii) the second one closer to the view of actor's system, relationship and functions. The former research field, hereby represented by works of Park and Sugumaran (Park *et al*, 2005) and Trabelsi, Ezzedine, Kolski (Trabelsi *et al*, 2004), is often software system oriented; for this reason the models derived from these studies are usually focused on relationship among agents, rather than on the role they assume. The latter approach is characterized by an higher abstraction level. The system's architecture and the agents role assume in this case more importance, as highlighted in the meta-model for MAS building presented in (Gomez *et al*, 2002). In both cases the ability of agents to solve problems is not considered an important characteristic. A methodology that brings in evidence the agent role assume a key importance in system design and it is a distinguishing point of agent oriented modeling over object oriented modelling; (Zambonelli *et al*, 2003) state that for complex systems, a clear distinction between the active actors of the systems and the passive resources may provide a simplified modelling of the problem. Moreover, delegating control to autonomous components can be considered as an additional dimension of modularity encapsulation

Our proposal for the development of models for MAS building emphasizes this characteristic by focusing on the Operations Research (OR) techniques in order to solve the local or global agent's problem. In the literature, there are several definitions for OR, sometimes dependent of application field. Hereby it is defined as a science that deploys scientific methods like mathematical modeling, statistics, and algorithms in order to make decisions in complex real-world problems. In this paper we propose two possible integrated approaches that can be used for create a MAS model: (i) the OR-based one uses algorithms to obtain optimal and shared solutions among agents while (ii) the second one is based on MAS traditional approach, more focused on architecture and relationship among system components. For the latter approach we suggest the Unified Modeling Language (UML) which is able to

represent the complex system with an object (agent) - oriented view: (Bauer et al, 2005).

### 2.2 The Reference Model

The reference model described in the Figure 1 consists of six steps: problem analysis, static architecture definition, multi-agent society definition, toolkit selection, model implementation and test & validation. The second and third steps have two different integrated implementations in order to put into evidence the importance of the agent's intelligence properties. The development of a multi-agent model is composed not only of the role and relationship's definition (left branch in Figure 1) but also of the investigation of the agent's ability to solve local problems using OR methods (right branch). Before to select the simulation's tool able to recreate the dynamics of real system, a model check is needed. At the end, the simulation step allows the verification of the impact that agent's decisions have on the achievement of the global objective.

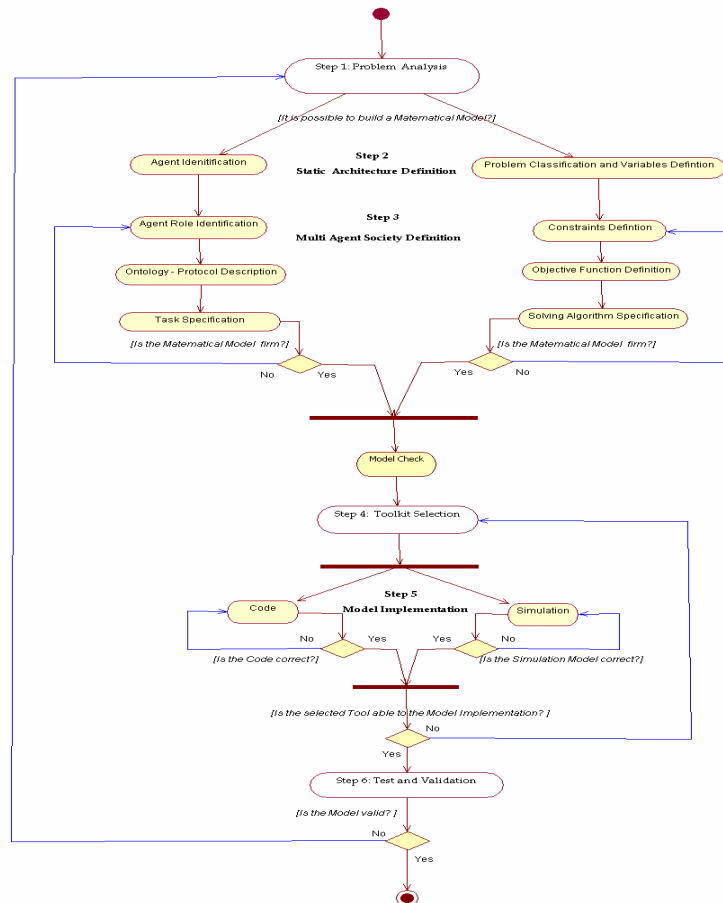


Figure 1 – The Reference Model

### 3. REFERENCE MODEL: AN APPLICATION TO FMS

In this section we describe how to realize a multi-agent model, with the aims of describing an application of the reference model presented in the previous section and validating it. In particular, the capabilities offered by an application to shoe flexible manufacturing system is investigated.

#### 3.1 Problem Analysis

The concept of Flexible Manufacturing System (FMS) was introduced in response to the following need: to improve the responsiveness to changes in products, production technology and markets. This issue has been discussed deeply in the literature (El Maraghy H.A., 2006). FMSs have high degree of complexity and often are underused mostly due to lack in software systems and communication technologies. Recently, new advances in wireless communication technologies and embedded systems boosted the possibility to implement organization, production planning and scheduling techniques and software tools able to exploit all the opportunities given by FMS. The Multi-agent models are often applied in FMS field (Adacher L. *et al*, 2000) (Hao Q. *et al*, 2003) as a promising technology for dealing with cooperation and decision-making in distributed application. In this work we consider an agile shoe manufacturing plant with innovative transportation lines. The innovative molecular structure of the transportation system allows the products to overtake along production lines, increasing the overall flexibility of the plant (Martinez Lastra J.L *et al*, 2006). In order to allow job overtake, it is important to analyze the job scheduling problem that is usual solved with the aim of optimizing different objectives, such as the makespan, the total cost, or penalty functions compliant with due dates.

##### 3.1.1. Domain Description

The domain can be described through the following components: (i) one Information System of the Plant that receives the orders demand from the market; (ii) six Manufacturing Cells (IA) processing customer orders. IA are linked by a flexible transportation line; (iii) a set consisting of three *customers*  $C_k$  ( $k = 1, \dots, 3$ ), in which each customer will issue one order demand with a total of three orders  $D_k$  ( $k = 1, \dots, 3$ ). Every order has a specific production time and a related due date.

#### 3.2 Static Architecture Definition

The second step of the reference model is the definition of the static architecture for describing the real entities or rather the intelligent agents operating in the system.

##### 3.2.1. Agents and variables identification.

Two types of agents are modeled. The Coordinator Agent (CA) who represents the Information System and the Island Agents (IA) who represent the manufacturing cells of the plant. Among the system variables we notice order due date and completion time for production and customer service level.

### 3.2.2 Characterization of the problem.

In order to respect or optimize the lead time for the customer, the agents must decide in which sequence to process the demand orders. In particular: (i) IAs decides the sequence of orders to be processed based on the related due date, (ii) CA, decides whether making changes to the released production plan.

## 3.3 Multi Agent Society Definition

An important feature of intelligent agents is their ability to communicate with other agents by sending messages and signals. In several works related to MAS theory the negotiation among competitive agents is often considered. Hereby the agents are cooperative actors that share local information to estimate the goodness of different solutions.

### 3.3.1 Role identification

CA plays a coordinator and strategic decision maker role, since it is the only agent able to communicate with all other agents and with external market. CA knows the latest information about the change of order's due date, or about the saturation degree of production plant capacity. The IAs have a functional role since they solve a local decision problem and carry out the decision made. They can communicate only with neighbor agents.

### 3.3.2 Protocol description

The information flow among various agents is regulated by the Overtake Coordination Protocol (OCP) that consists of the following steps:

1. The CA receives orders from the market and communicate to the first IA the release date and due date of orders.
2. The first IA solves a local scheduling problem and communicate to the next IA when it can start to process the orders.
3. If the due date of an order changes during production or distribution process, then the CA communicate the new due date to the IA following the IA who is working the changed order.
4. The IA finds the best schedule and sends to CA this schedule with ending times for every order.
5. The CA calculates the convenience of changing schedule and, if overtaking is convenient, sends the order of overtaking to the IA.

### 3.3.3 Tasks specification

Each IA solves a *scheduling problem* by determining the best sequence of order processing based on the related due date. The global shared goal is that the system must minimize the order delays for maximizing the customer service level. At more high level, CA decides whether allowing to overtake the order minimizes the following decisional function introduced in (Baffo I. *et al*, 2007):

$$f = \alpha * \Delta T + \beta * \Delta C_{max} \quad (1)$$

where  $\alpha$  is a parameter that depends on order priority and  $\beta$  is a parameter depending on how many order must be processed again;  $\Delta T$  is the change in delay of the order while  $\Delta C_{max}$  is the change in all orders completion time after that the order schedule is changed.

### 3.3.4 Constraints definition

In the problem solving process, the following constraints must be respected: (i) all orders must be processed; (ii) every order must respect the precedence constraints.

### 3.3.5 Objective function definition

For scheduling problem of the IAs, the objective function is to minimize the lateness like difference between the ending time of every order and their due dates. The problem is known in literature like  $1|r|L_{max}$ , where the letter  $r$  represents the possibility to consider the preemptive solution. The CA goal is to maximize the service level (SL), the decision is made using function (1).

### 3.3.6 Solution Algorithm Definition

The IA's scheduling problem is solved through a branch and bound based algorithm.

### 3.3.7 Model Check

The check of the model is made verifying the correctness of logic relationship among the most important parameters of the problem. Changing the value of some parameters, the output of the model correspond to the output of the real system, with a approximation due to the impossibility to consider all variables of the problem.

## 3.4 Toolkit Selection

The selection of the toolkit depends on factors such as the problem size, the quality of solution, the number of solutions that can be evaluated, the available time for solving the problem. For the test case presented in this section the Java programming language is used to build the model for the manufacturing and distribution network.

## 3.5 Test and Validation

Test and validation have been performed in order to show how the model can affect the flexibility performance. We present two exemplificative scenarios of change in order due date while the production and distribution plan is released. In the first scenario the due date of order 1 changes after the first IA. In the second one the due date of order 3 changes after the third IA. In the first case, if we assume that  $\alpha$  is the same of  $\beta$ , then the CA is willing to change the schedule because the worsening of the  $C_{max}$  is smaller than the advantages of reducing the tardiness of order 1. In this case the external flexibility in term of *customer service level*, meant as the ability to respond quickly to market change, is more important than the completion of all the orders. In the second scenario, if we assume that  $\alpha$  is the same of  $\beta$ , the CA is willing not to change the schedule, because the advantage of reducing the tardiness of order 3 is smaller than the worsening of the  $C_{max}$ . In this case, the *internal flexibility*, meant as the ability to manage in efficient way the plant, is more important than responding quickly to market requests. In both cases the values of  $\alpha$  and  $\beta$  determine the decision of CA.

#### 4. REFERENCE MODEL VALIDATION

The reference model validation is reported in terms of applications to several problem to different fields. In particular we report two applications: (i) one related to a *Vehicle Routing Problem with Time Window* (VRPTW) in a distribution process faced in (Bianco *et al*, 2005); (ii) one related to a *Collaborative Network* in an industrial local system (Baffo I. *et al*, 2007). In the first problem, we identify three types of agents: (i) a Logistic Operator (LO), involved in the delivery of goods from a single depot to customers while meeting customer service level requirements; (ii) the Truck Operator (TO), responsible for the delivery of goods by means of own vehicles and with the aim of minimizing the total traveled distance; (iii) the Customer (C) that has to receive goods within the requested delivery time. The building of a MAS through the presented reference model allowed to obtain a good solution as trade-off between the penalty cost linked to delivery out of time windows and the route distance covered by TO. Moreover the solution proposed can be considered as an output of a negotiation process between the LO and TO. C in this case is an information possessor because the time windows supplied by C cannot be changed by the system dynamics. In the second application proposed, the problem is related to the allocation of project's activities to industrial agent operating in a geographic area. To every activity corresponds a profit and the scope of local enterprises is to maximize the total profit obtaining a great number of activities. The system's agents are: (i) *one* coordinator agent (CA) that decides in which way to allocate the project resources on based of enterprise's competence. (ii) *n* industrial agents that declare their competence level and during the time invest to increase this level. The solutions of these problems have allowed to measure the results in terms of efficient allocation of the available resources and in terms of system evolution towards growing level of global competence on the territory.

#### 5. CONCLUSION

In this paper we presented a reference model able to support a designer in building multi-agent models for representing complex systems. In particular we provided guidelines in order to create a model in which: (i) the intelligent agents solve local problems with OR techniques; (ii) a shared solution of the problem at global level is obtained through the communications and collaboration among different plant agents. The applications performed in supply chain and industrial district fields, proved the validity and ability of the study as innovative decision support system that can be designed under a multi-agent point of view. The test cases proved as a well designed model can be applied with a few changes to different environments. Furthermore, the experiments validated the robustness of the reference model with respect to the field of application, type of problem and solution methods.

## 6. REFERENCES

1. Adacher L., Agnetis A., Meloni C., 2000, Autonomous agents architectures and algorithms in flexible manufacturing systems, *IIE Transactions* Vol32, 941-951.
2. Baffo I., Confessore G., Rismondo S., Uno strumento di modellazione e simulazione della dinamica di una rete di attori operanti in un territorio, in proceedings of XXVIII Italian Conference of Regional Science, Bolzano, 2007.
3. Bauer B, Odell J. UML 2.0 and agents: how to build agent based systems with the new UML standard. *Engineering Application of Artificial Intelligence*, 2005; 18: 141-157.
4. Bianco L, Confessore G, Stecca G, A multi-agent model for distribution problems in logistics systems. In Proceedings of 18th International Conference on Production Research ICPR-18: The networked enterprise: a challenge for a sustainable development, 2005; CD-ROM.
5. Camarinha-Matos LM, Afsarmanesh H. A modeling framework for collaborative networked organizations, in IFIP International Federation for Information Processing, volume 224, Network-Centric Collaboration and Supporting Fireworks, eds. Camarinha –Matos, L., Afsarmanesh, H., Ollus, M., (Boston: Springer), 2006; 3-14.
6. Camarinha-Matos LM, Afsarmanesh H. Collaborative networks: A new scientific discipline. *Journal on Intelligent Manufacturing*, 2005; 16: 439-452.
7. El Maraghy H.A., 2006, Flexible and reconfigurable manufacturing systems paradigms, *International Journal of Flexible Manufacturing Systems*, Vol17, 261–276.
8. Gomez-Sanz, JJ, Pavon, J, Garijo F. Metamodels for building multi-agent systems, in Proceedings of the 2002 ACM symposium on Applied computing. ACM Press, 2002; 37–41.
9. Hao Q., Shen W., Stecca G., Wang L., 2003, Towards an internet enabled cooperative manufacturing management framework, *Processes and Foundations for Virtual Organizations*, L.M. Camarinha-Matos and H. Afsarmanesh (eds.), 191-200.
10. Jennings NR, Wooldridge M. Applying agent technology. *Applied Artificial Intelligence*, 1995; 9: 357-369.
11. Martinez Lastra J.L., Colombo A.W., 2006, Engineering framework for agent-based manufacturing control, *Engineering Applications of Artificial Intelligence*, Vol19, 625–640.
12. Park S, Sugumaran V. Designing multi-agent systems: a framework and application. *Expert Systems with Applications*, 2005; 28: 259-271.
13. Trabelsi A, Ezzedine H, Kolski C. Architecture modelling and evaluation of agent based interactive system. *IEEE International Conference on Systems, Man and Cybernetics*, 2004
14. Zambonelli F, Jennings NR, Wooldridge M. Developing Multiagent Systems: The Gaia Methodology. *ACM Transaction on Software Engineering and Methodology*. Vol 12, No. 3. July 2003, Pages 317-370.