# UNIVERSITÀ DEGLI STUDI DI ROMA "TOR VERGATA"

FACOLTA' DI INGEGNERIA

DOTTORATO DI RICERCA IN INGEGNERIA ECONOMICO GESTIONALE


XXII CICLO

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


Ilaria Baffo


A.A. 2008/2009


Docente Guida/Tutor: Prof. Giuseppe Confessore

Coordinatore: Prof. Agostino La Bella

Co-Tutor: Ing. Giuseppe Stecca

# ACKNOWLEDGMENT

I would like to express my gratitude to all those who have made this thesis possible.

*"E' meno male l'agitarsi nel dubbio che il riposar nell'errore"*

It's better to fret in doubt than to rest in error

Alessandro Manzoni

*"Io mi dico è stato meglio lasciarci*
*che non esserci mai incontrati."*

Fabrizio De Andrè

(Giugno '73)

# ABSTRACT


*During the last years the decision making processes is evolving becoming more and more distributed and asynchronous. In order to support decision-makers who are not at the same place at the same time are defined cooperation processes and a set of models able to support designers of Cooperative decision support framework. One of this, proposed in this thesis, is a Reference Model to build Multi Agent System able to represent complex and distributed system composed by several intelligent collaborative entities. Following this definition, it is possible define like collaborative network a lot of complex system such as:*

1. *Flexible Manufacturing Systems (FMSs) that are defined by Ciliers (Ciliers P., 1998) as complex systems in which the knowledge of the system elements composition and the interaction between the environment and the system, is not enough to understand the system functioning;*
2. *Industrial Clusters (ICs) that can be defined as socioeconomic entities characterized by a social community of people and a population of economic agents localized in close proximity in a specific geographic region that interact to reach similar outcomes (Marshall, 1925);*
3. *Supply Chains (SCs), network of suppliers, factories, warehouses, distribution centres and retailers that operate in integrated manner (Fox, 1992) with the same aim of reducing inventory and costs, adding value, extending resource and accelerating the time to market.*
4. *Healthcare Systems, composed by several entities interacting in a great number of critical processes. The internal dynamics of an hospital represents a complex non-linear structure hard to manage in centralized way (Harper 2002).*

*The above mentioned list represents only a part of the numerousness examples of* *system that can be represented as collaborative networks and that need to be managed in distributed manner due to complexity and to great amount of information necessary to take any kind of decision. There are not many models able to represent and manage this kind of distributed systems, Petri nets represent one of this but present some limits due to: (i) the model become too large and complex even for a modest size problem (Wang J., Deng Y., 1999); (ii) the difficulty to make even a little change to a previously built model.*

*The Multi Agent System (MAS) approach aim to overcome these limits of rigidity and computing complexity trough: (i) the possibility to solve complex problem solving a set of easier local problems; (ii) the opportunity to change some problem parameters or to substitute any system element without discard entirely the original model. The MAS are widely studied in literature like method able to represent dynamic and distributed system with several decision makers having different information domains. It is possible to observe a lack in investigate the problem solving ability of intelligent agents in a multi-agent setting and a variety of representation methods. The difficulty of apply this model approach reside into the activity of architecture design used by the system elements to speak and act. The number and the kind of relations among the system entities became the most important indicator of the model's complexity. For this reason the outcome of my researches can be presented as a reference model integrating two types of approach for the models creation in the MAS field. The main scope is to provide the guidelines able to support the designer in the system modelling like MAS, indicating also in which scenario is more convenient to adopt an approach oriented to Operation Research (OR) technique or another one. The Reference Model is evaluated and validated thanks to several application in different contests, some of whom are mentioned at the beginning of this abstract.*

# CONTENTS

# 1

# INTRODUCTION

## 1.1 Decision Support Systems

Decision Support Systems (DSSs) are born in the 70's.

Decision-makers must take into account a lot of information coming from different entities and have sometimes a large amount of information to aggregate in order to reach the best solution. They need a personal and direct control on systems, which are designed for them. Decision and Information have to be classified referring to the level of management. We could distinguish three levels of management: operational level, control level and strategic management level.

According to Sprague and Carlsson (Sprague & Carlsson, 1982) at the operational level, information is numerous and very detailed. The same phenomenon could be observed for decision: there are a lot of decisions to make at the operational level and information usable for them is also in great proportion. At the control level, information is less numerous than in the previous case and decisions to make are more important. Information at this level is more aggregated. At the strategic management level, information is aggregated and not numerous and decisions to make have a very high impact in the management.

Decision-makers have to be supported in their task by systems like Decision Support Systems regardless of the level of management. The system will integrate and present information aggregated or not depending of the management type.

Secondly, it is always possible to distinguish two kinds of decisions: non structured decisions or semi-structured decisions and well structured decisions. In the first case decision could be described by rules and procedures and then they could be programmed and performed by a system. In the case of non-structured decision, Decision Support Systems are useful.

## 1.2 From DSS to Cooperation Model

Definitions of Decision Support Systems are very numerous. One of the most known definition has been introduced by Keen and Scott Morton (Keen and Scott Morton , 1978): DSS imply computers use for (1) support decision makers in their decision process for semi-structured tasks, (2) help rather than replace decision makers judgement, (3) improve decisions effectiveness rather than efficiency. This definition points out the fact that DSSs are included in the decision process and that efficiency is no more the central objective. Another definition completes the previous one and has been introduced by Sprague and Carlsson (1982): DSS could be characterised as interactive computer systems that support decision makers by using data and models to solve unstructured problems. This definition introduces another dimension of DSSs: *interactivity*. The solution is found interactively between the user and the system. The user will stop the solving problem process when he will find his satisfactory solution. The interactivity gives to DSS a cooperative dimension. The relation between the system and the user is a cooperative relation because it implements a particular mode of cooperation that could be seen as a specific task shared by both the system and the user. Faced with rapid changes resulting from information and communication technologies (ICT), many organisations enlarge the decision process. *Different decision-makers are implied in the decisional process*.

They work together but not necessarily at the same time and at the same place. We could define this kind of process as: asynchronous distributed decision making or sharable decision making or more simply Distributed Decision Making.

In order to support this kind of decisional process, Cooperative Decision Support Framework must be designed. Cooperative decision support framework must be able to support cooperative activity in a distributed asynchronous decision making way.

## 1.3 Cooperative System

According to Soubie (Soubie, 1998) the cooperation could be defined as follows:

• a common goal,

• actors,

• communication tools,

• tasks to reach the common goal have to be sharable.

The cooperation mode is revisable during the process of problem solving according to the context evolution. Each actor could have a local objective. If this last comes to contrary the common goal, it is then a particular form of cooperation that is negotiation. Negotiation can be viewed as the lowest level of cooperation on a continuous scale of cooperation, corresponding to a lack of representation of mutual intervention regarding the common goal. The main aim of this work is to develop a framework able to support distributed decision-making. The decision support framework must then include a cooperative component in order to manage the different decision makers to reach a common goal.

A cooperative decision support framework must fulfil the following requirements:

• support the different decision-makers involved in the asynchronous and distributed

• include a communication tool;

• include a task editor that allows users to decompose tasks in sub-tasks and to assign tasks to decision-makers.

Decision-making could be seen as a particular problem solving case. Cooperative decision support framework is based on the problem solving modelling and on the cooperation modelling. The modelling step in the cooperative decision support framework design process takes a great importance.

.

# 2

# MULTI AGENT SYSTEM

## 2.1 Introduction

Most researchers in AI to date have dealt with developing theories, techniques, and systems to study and understand the behaviour and reasoning properties of a single cognitive entity. AI has matured, and it endeavours to attack more complex, realistic, and large-scale problems. Such problems are beyond the capabilities of an individual agent. The capacity of an intelligent agent is limited by its knowledge, its computing resources, and its perspective. This bounded rationality is one of the underlying reasons for creating problem-solving organizations. The most powerful tools for handling complexity are modularity and abstraction. Multi-agent systems (MASs) offer modularity. If a problem domain is particularly complex, large, or unpredictable, then the only way it can reasonably be addressed is to develop a number of functionally specific and (nearly) modular components (agents) that are specialized at solving a particular problem aspect.

This decomposition allows each agent to use the most appropriate paradigm for solving its particular problem. When interdependent problems arise, the agents in the system must coordinate with one another to ensure that interdependencies are properly managed.

Furthermore, real problems involve distributed, open systems (Hewitt 1986). An open system is one in which the structure of the system itself is capable of dynamically changing. The characteristics of such a system are that its components are not known in advance; can change over time; and can consist of highly heterogeneous agents implemented by different people, at different times, with different software tools and techniques. Perhaps the best-known example of a highly open software environment is the internet. The internet can be viewed as a large, distributed information resource, with nodes on the network designed and implemented by different organizations and individuals. In an open environment, information sources, communication links, and agents could appear and disappear unexpectedly. Currently, agents on the internet mostly perform information retrieval and filtering.

The next generation of agent technology will perform information gathering in context and sophisticated reasoning in support of user problem-solving tasks. These capabilities require that agents be able to interoperate and coordinate with each other in peer-to-peer interactions. In addition, these capabilities will allow agents to increase the problem-solving scope of single agents. Such functions will require techniques based on negotiation or cooperation, which lie firmly in the domain of MASs (Jennings, Sycara, and Wooldridge 1998; O'Hare and Jennings 1996).

It is becoming increasingly clear that to be successful, increased research resources and attention should be given to systems consisting of not one but multiple agents. The distributed AI (DAI) community that started forming in the early 1980s and was tiny compared to mainstream, single-agent AI is rapidly increasing.

The growth of the MAS field is indisputable. Research in MASs is concerned with the study, behaviour, and construction of a collection of possibly pre-existing autonomous agents that interact with each other and their environments. Study of such systems goes beyond the study of individual intelligence to consider, in addition, problem solving that has social components. An *MAS* can be defined as a loosely coupled network of problem solvers that interact to solve problems that are beyond the individual capabilities or knowledge of each problem solver (Durfee and Lesser 1989).

These problem solvers, often called *agents*, are autonomous and can be heterogeneous in nature. The characteristics of MASs are that:

1. each agent has incomplete information or capabilities for solving the problem and, thus, has a limited viewpoint;
2. there is no system global control;
3. data are decentralized;
4. computation is asynchronous.

The motivations for the increasing interest in MAS research include the ability of MASs to do the following.

*First* is to solve problems that are too large for a centralized agent to solve because of resource limitations or the sheer risk of having one centralized system that could be a performance bottleneck or could fail at critical times.

*Second* is to allow for the interconnection and interoperation of multiple existing legacy systems. To keep pace with changing business needs, legacy systems must periodically be updated. Completely rewriting such software tends to be prohibitively expensive and is often simply impossible. Therefore, in the short to medium term, the only way that such legacy systems can remain useful is to incorporate them into a wider cooperating agent community in which they can be exploited by other pieces of software. Incorporating legacy systems into an agent society can be done, for example, by building an agent wrapper around the software to enable it to interoperate with other systems (Genesereth and Ketchpel 1994).

*Third* is to provide solutions to problems that can naturally be regarded as a society of autonomous interacting components-agents. For example, in meeting scheduling, a scheduling agent that manages the calendar of its user can be regarded as autonomous and interacting with other similar agents that manage calendars of different users (Garrido and Sycara 1996). Such agents also can be customized to reflect the preferences and constraints of their users. Other examples include air-traffic control (Kinny et al. 1992) and multi-agent bargaining for buying and selling goods on the internet.

*Fourth* is to provide solutions that efficiently use information sources that are spatially distributed. Examples of such domains include sensor networks, seismic monitoring, and information gathering from the internet (Sycara et al. 1996).

*Fifth* is to provide solutions in situations where expertise is distributed. Examples of such problems include concurrent engineering (Lewis and Sycara 1993), health care, and manufacturing.

*Sixth* is to enhance performance along the dimensions of (1) *computational efficiency* because concurrency of computation is exploited (as long as communication is kept minimal, for example, by transmitting high-level information and results rather than low level data); (2) *reliability*, that is, graceful recovery of component failures, because agents with redundant capabilities or appropriate inter-agents coordination are found dynamically (for example, taking up responsibilities of agents that fail); (3) *extensibility* because the number and the capabilities of agents working on a problem can be altered; (4) *robustness*, the system's ability to tolerate uncertainty, because suitable information is exchanged among agents; (5) *maintainability* because a system composed of multiple components-agents is easier to maintain because of its modularity; (6) *responsiveness* because modularity can handle anomalies locally, not propagate them to with different abilities can adaptively organize to solve the current problem; and (8) *reuse* because functionally specific agents can be reused in different agent teams to solve different problems.

MASs are now a research reality and are rapidly having a critical presence in many human-computer environments.

### 2.1.1 Multi Agent System Issues and Challenges

Although MASs provide many potential advantages, they also present many difficult challenges. Here, I present problems inherent in the design and implementation of MASs. The list of challenges includes problems such as:

First, how do we formulate, describe, decompose, and allocate problems and synthesize results among a group of intelligent agents?

Second, how do we enable agents to communicate and interact? What communication languages and protocols do we use? How can heterogeneous agents interoperate? What and when can they communicate? How can we find useful agents in an open environment?

Third, how do we ensure that agents act coherently in making decisions or taking action, accommodating the nonlocal effects of local decisions and avoiding harmful interactions? How do we ensure the MAS does not become resource bounded? How do we avoid unstable system behaviour?

Fourth, how do we enable individual agents to represent and reason about the actions, plans, and knowledge of other agents to coordinate with them; how do we reason about the state of their coordinated process (for example, initiation and completion)?

Fifth, how do we recognize and reconcile disparate viewpoints and conflicting intentions among a collection of agents trying to coordinate their actions?

Sixth, how do we engineer and constrain practical DAI systems? How do we design technology platforms and development methodologies for MASs?

Solutions to these problems are intertwined (Gasser 1991). For example, different modelling schemes of an individual agent can constrain the range of effective coordination regimes; different procedures for communication and interaction have implications for behavioural coherence. Different problem and task decompositions can yield different interactions. It is arguable whether one can find a unique most important dimension along which a treatment of MASs can cogently be organized. Here, I attempt to use the dimension of effective overall problem-solving coherence of an MAS as the organizing theme. Ensuring that an MAS exhibits coherent collective behaviour while it avoids unpredictable or harmful behaviour (for example, chaos, oscillation) is indeed a major challenge: By its very nature, an MAS lacks global perspective, global control, or global data. Coherence is a global (or regional) property of the MAS that could be measured by the efficiency, quality, and consistency of a global solution (system behaviour) as well as the ability of the system to degrade gracefully in the presence of local failures. Several methods for increasing coherence have been studied. These methods, along with issues of single-agent structuring in an MAS, cover the topics I want to survey here.

### 2.1.2 Individual Agent Reasoning

Sophisticated individual agent reasoning can increase MAS coherence because each individual agent can reason about nonlocal effects of local actions, form expectations of the behaviour of others, or explain and possibly repair conflicts and harmful interactions. Numerous works in AI research try to formalize a logical axiomatization for rational agents (see Wooldridge and Jennings, 1995) for a survey).

This axiomatization is accomplished by formalizing a model for agent behavior in terms of beliefs, desires, goals, and so on. These works are known as belief-desire-intention (BDI) systems (Shoham, 1993). An agent that has a BDI-type architecture has also been called *deliberative.*

In my own work on multi-agent infrastructure, agents coordinate to gather information in the context of user problem solving tasks. Each agent is a BDI-type agent that integrates planning, scheduling, execution, information gathering, and coordination with other agents (Sycara et al. 1996). Each agent has a sophisticated reasoning architecture that consists of different modules that operate asynchronously. The *planning module* takes as input a set of goals and produces a plan that satisfies the goals. The agent planning process is based on a hierarchical task network (HTN) planning formalism. It takes as input the agent's current set of goals, the current set of task structures, and a library of task-reduction schemas. A *task-reduction schema* presents a way of carrying out a task by specifying a set of subtasks-actions and describing the information-flow relationships between them. The *communication and coordination module* accepts and interprets messages from other agents. Messages can contain requests for services. These requests become goals of the recipient agent. The *scheduling module* schedules each of the plan steps. The agent scheduling process takes as input, in general, the agent's current set of plan instances and, in particular, the set of all executable actions and decides which action, if any, is to be executed next. This action is then identified as a fixed intention until it is actually carried out (by the execution component). Agent-reactivity considerations are handled by the execution-monitoring process. Execution monitoring takes as input the agent's next intended action and prepares, monitors, and completes its execution. The *execution monitor* prepares an action for execution by setting up a context (including the results of previous actions) for the action. It monitors the action by optionally providing the associated computation-limited resources—for example, the action might be allowed only a certain amount of time, and if the action does not complete before the time is up, the computation is interrupted, and the action is marked as having failed. Failed actions are handled by the *exception-handling process.*

The agent has a domain-independent library of plan fragments (task structures) that are indexed by goals as well as a domain-specific library of plan fragments that can be retrieved and incrementally instantiated according to the current input parameters.

Reactive agents have also been developed.

Reactive agents have their roots in Brooks's (1991) criticism of deliberative agents and his assertions that (1) intelligence is the product of the interaction of an agent and its environment and (2) intelligent behaviour emerges from the interaction of various simpler behaviours organized in a layered way through a master-slave relationship of inhibition.

A different reactive architecture is based on considering the behaviour of an agent as the result of competing entities trying to get control over the actions of the agent. An *agent* is defined as a set of conflicting tasks where only one can be active simultaneously. A *task* is a high-level behavioural sequence as opposed to the low-level actions performed directly by actuators. A reinforcement mechanism is used as a basic learning tool to allow the agents to learn to be more efficient in tasks that are often used.

### 2.1.3 Organization of Agent Society

An *organization* provides a framework for agent interactions through the definition of roles, behaviour expectations, and authority relations. Organizations are, in general, conceptualized in terms of their structure, that is, the pattern of information and control relations that exist among agents and the distribution of problem solving capabilities among them. In cooperative problem solving, for example (Corkill and Lesser 1983), a *structure* gives each agent a high level view of how the group solves problems. The structure should also indicate the connectivity information to the agents so they can distribute sub-problems to competent agents.

In open-world environments, agents in the system are not statically predefined but can dynamically enter and exit an organization, which necessitates mechanisms for agent locating. This task is challenging, especially in environments that include large numbers of agents and that have information sources, communication links, and/or agents that might be appearing and disappearing. Researchers have identified different kinds of middle agent that help agents find others. When an agent is instantiated, it advertises its capabilities to a middle agent. An agent that is looking to find another that possesses a particular capability (for example, can supply particular information or achieve a problem-solving goal) can query a middle agent. In an agent infrastructure, there could be multiple middle agents, not only in type but also in number. Another perspective in DAI defines organization less in terms of structure and more in terms of current organization theory. For example, Gasser (1991) views an organization as a "particular set of settled and unsettled questions about beliefs and actions through which agents view other agents." In this view, an organization is defined as a set of agents with mutual commitments, global commitments, and mutual beliefs. An organization consists of a group of agents, a set of activities performed by the agents, a set of connections among agents, and a set of goals or evaluation criteria by which the combined activities of the agents are evaluated. The organizational structure imposes
constraints on the ways the agents communicate and coordinate. Examples of organizations that have been explored in the MAS literature include the following:

**Hierarchy:** The authority for decision making and control is concentrated in a single problem solver (or specialized group) at each level in the hierarchy. Interaction is through vertical communication from superior to subordinate agent, and vice versa. Superior agents exercise control over resources and decision making.

**Community of experts:** This organization is flat, where each problem solver is a specialist in some particular area. The agents interact by rules of order and behaviour (Lewis and Sycara 1993). Agents coordinate though mutual adjustment of their solutions so that overall coherence can be achieved.

**Market:** Control is distributed to the agents that compete for tasks or resources through bidding and contractual mechanisms. Agents interact through one variable, price, which is used to value services (Mü llen and Wellman, 1996). Agents coordinate through mutual adjustment of prices.

**Scientific community:** This is a model of how a pluralistic community could operate. Solutions to problems are locally constructed, then they are communicated to other problem solvers that can test, challenge, and refine the solution (Lesser 1991). In open, dynamic environments, the issue of organizational adaptivity is crucial. Organizations that can adapt to changing circumstances by altering the pattern of interactions among the different constituent agents have the potential to achieve coherence in changing and open environments. A multi agent system can exhibit organizational adaptivity through cooperation mediated by middle agents. The model's agents built with the reference model find their collaborators dynamically based on the requirements of the task and on which agents are part of the society at any given time, thus adaptively forming teams on demand.

## 2.2 Distributed Decision Making on Manufacturing System

During the last years, the development of the new technologies – and of the “organization and management science” - has conditioned deeply the structure and the operation of the production systems. Such change is visible above all in the manufacturing factories.

In fact, the electronic and computer science, mechanical technologies and their integration have allowed to the automation of many production processes and contextually, to the attainment of a great number of benefits, which for example, minors production costs, greater quality of product, and more elevated degree of flexibility.

The flexibility term means the degree of quickly adaptation to the market's variability, that is the ability to change the amount and the type of product following the demand.

In parallel with the technological development, it has been understood that in order to use in optimal way the automatic systems, it was necessary to development:

- a new organization approach,
- an ability to face the problems in global and "integrated" way and
- a new method to represent the Factory System.

Moreover the increase and the diversification of the products demands from the market have show the inefficiency of a rigid organization system, in respect of a more flexible structure as that composed by more functions able to realize specific tasks.

This structure therefore leads to the problems connected with the exchange of the information and the ability to adapt the single function's capabilities to the market needs. (re-configurability).

In particular it is possible represent the factory and the informative flow in the following way:

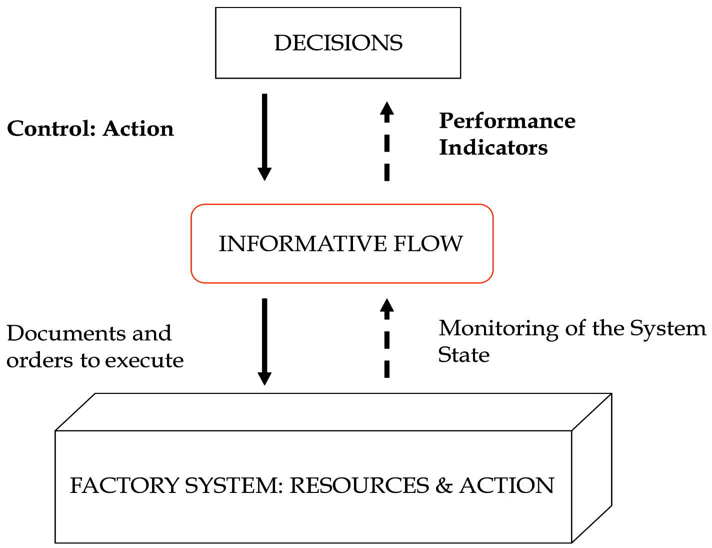


**Figure 1: Conceptual scheme of the company**

Inside of the factory it is possible to distinguish a great set of resources, understandings as what it is necessary in order to carry out the production process, and that participates in it, enduring or producing transformations. They can be, physical, informative, financial, human, and temporal resources.

Moreover, the productive process consist in action that operate on these resources.

The actions can consume, use or transform the resources.

The interaction between the actions and the resources it is possible through a bidirectional informative flow.

In a direction, the flow records and measure the actual state of system's resources, and in another direction, influence this state through the documents release (production orders, project orders, etc…) that realize the same action.

Such state of information constitutes the interface between the physical reality of the company and the decisional level. In fact, the great part of decision in manufacturing environment, are taken on the basis of system acquaintance that it is had through its representation. In this level, it is collocate the Modelling Activity, that must supply the correct representation of more important variables, and their interaction, of the production system. In order that the decisions taken can be effective, it is important that the actual state of the system and the data that describe are consistent.

Obviously, it is completely unthinkable to construct only a model able to include all decisional aspects of a manufacturing system. This, not only because the problem would be too much complex, but, above all, because the decisional process has a distributed nature that generates various difficulties due to the coordination of the multiple decision-makers. For these reasons, often, it is usual *decompose* the general problem in more sub-problem, whose resolution is less difficult.

Naturally, whit this approach, the sum of single local optimal will be different from the global optimum; they are disadvantages about the goodness of the general solution, but advantages about the time and the facility of resolution of the problem.

### 2.2.1 The Decision Problems

It is possible subdivide the aforesaid management problems in 2 classes:

- Planning Problem
- Scheduling Problem

*Planning Problem:* it regard the material flow management inside of the system, and their resolution must be made before that the production begins.

This type of problem include the issues about:

1. Part type selection
2. Machine grouping
3. Production ratios determination (definition of production mix)
4. Tooling (allocation of the tools)
5. Routing o Loading (assignment of the operations to the machine)

*Scheduling Problem*: it regard the determination of the best sequence with which introducing pieces in the system and with which processing pieces on the various machines. This type of problem are usual solved aiming to the optimization of various objectives, which for example, the makespan, or the total cost, or the penalties function related to the respect of due date.

Being these problems much complex, often they are solved through Euristic Methods, able to supply a good solution, but not optimal, in short real time.

### 2.2.2 Possible Approaches

Like saying in the introductory part of this job, from the technological point of view the Flexible Manufacturing System represent an instrument whit great potentialities in manufacturing environment. In the past, however, the application of these systems to the factories has not lead to the attended results. The reason of this inefficiency often derives from the insufficient attention that is given to the organizational and managerial problems.

For example, it is impossible to think to take advantage of the potentialities of the new machine's technology, without one planning of the time, a coordination and a balance of the workload of the such machines. Therefore in parallel with the increment of machine's productivity it has grow the need to development a methodologies in order to *formulate* and to *resolve* a great set of decisional problems that before did not exist. To these two fundamental steps correspond two conceptual activities, famous as *modelling activity* and *algorithmic activity*. The first consists in representing the productive system and the decisional problem with opportune mathematical instruments. The second one consists in finding the more convenient way (not always optimal) to resolve a decisional problem.

It is useful to have a set of instrument able to:

- represent in coherent way the real system and their decision problems
- resolve such problems
- estimate the choices carried out.

In literature exist various methods that include these instruments, and that can be group in two main traditions: one that faces the problems in traditional and centralized way, and one that proposes a new and distributed approach.

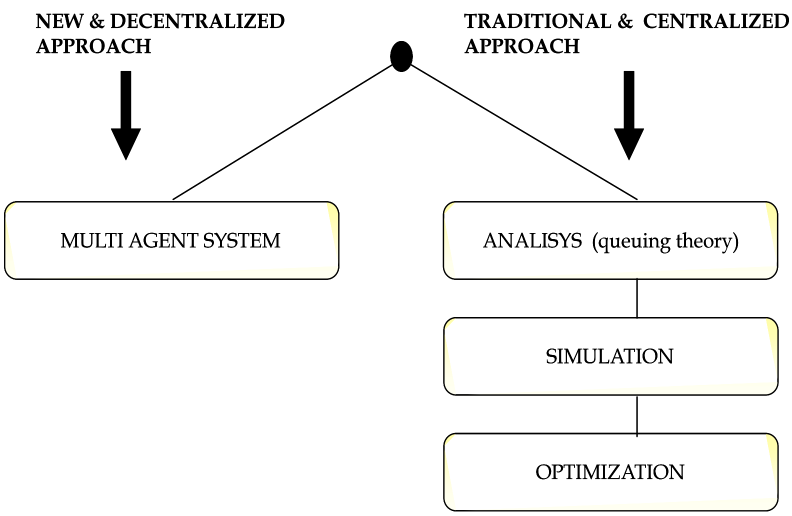


**Figure 2: Main approaches to the decisional problem**

## Traditional & Centralized Approach

*Simulation Models*

The simulation is an instrument that concurs to associate objects of the real world to byte. In this way, for example, moving this byte from a lease to the other of the memory, the computer simulates the movement of pieces from a working station to the other.

Therefore, the simulation model reproduce the system's behaviour, allowing to observe the parameter's values most important. However, so that a simulation model turns is necessary a great amount of data. Usually it is difficult to obtain these information, and their treatment can require high costs of time and resources. This approach is used when it is necessary a high detail degree.

*Analytical models*

Often the information on which the simulation models are based are not available or they are not necessary for solve specific problems. In this case, it is better adopt the analytical models, that through an analysis of principle of the system they allow to supply one solution in short times. These models use the methodologies of queuing theory and allow to observe in precise way the main variations and interrelations of the most important variables.

Regarding the first introduced models, these are more flexible regarding the variations in the parameter's values, while they are less one regarding the structural variations of the model.

A disadvantages of this approach is that it is necessary to make hypothesis that they could be revealed no realistic, an advantage is that the model needs of a small amount of data.

*Optimization Model*

An optimization model is an instrument of system's representation oriented to the solution of the problems: that is, it aiming to the optimization of an objective in the respect of a set of constraints.

One of the main advantages of this approach is that the construction of a optimization model forces to carry out a close analysis on the system's operation.

Like previously said, the dimensions of the problem can difficultly be faced from the systems of calculation currently available. In this case, the optimization model must *decompose* the total decisional problem in a whole structured of sub-problem tractable, such that the composition of their solutions supplies a good solution of the original problem.

**New & Traditional Approach**

*Multi Agent System*

The concept of Autonomous Agent was born in computer science. In literature many definitions of term "autonomous agent" are present, the most known definition is given by M. Wooldridge and N.R. Jennings, for which an autonomous agent is an entity, placed in a particular environmental, able of an independent behaviour and flexible actions to reach its own objective.

The key concepts of this definition are:

1. *positioning*, agent receives and answers to the external environment's stimulus.
2. *autonomy*, agent is in a position to act without the participation of other entities, for example a man or an other agent. Moreover it has control on its actions and on its inner state. Often it is referred to the ability of agents to learn from the experience (learning).
3. *flexibility*, the agent is:
   - reactive: he perceives and answers, in reasonable time, to the external stimulus.

- proactive: he is able to exhibit an opportunistic behaviour oriented to the pursue its own objective and act autonomously when needed.
- social: he interact with other agents and humans in order to succeed to pursue their objective and to help others to pursue theirs.

One of the fundamental aspects that characterize an agent is that social behaviour. One of its main characteristics is, indeed, the capacity to communicate with other agents and the surrounding environment , in order to create one Agents Society.

In a Multi Agent System more agents are present and they interact among them for the resolution of one or more problems. They are often used in order to represent problems that have one or more of this following characteristic:

- ✓ multiple Problem Solving Methods;
- ✓ multiple decision makers;
- ✓ multiple alternative activities.

A Multi Agent System's benefits are:

- ➢ distributed and concurrent problem solving: every agent has a problem to resolve, with a set of constraints to respect and a set of performance's indicators to maximize.
- ➢ Interaction model: every agent exchange messages with other entities increasing the total surplus through:
    - the cooperation: to work together for an single objective;
    - the coordination: to organize activity of problem solving avoiding negative interactions and taking advantage of positive interaction;
    - the negotiation: to arrive to a shared agreement in a competitive environment.

Therefore, a Multi Agent System can be defined as a set of agents that work together in order to resolve a problem that goes beyond the abilities and the knowledge of the single agent.

The characteristics of Multi Agent System are:

- every agent has an incomplete information of the situation in which it is integrated, or an incomplete ability to resolve the problem, therefore a limited point of view;
- a centralized control system may not exist;
- information and data of the problem are decentralized.

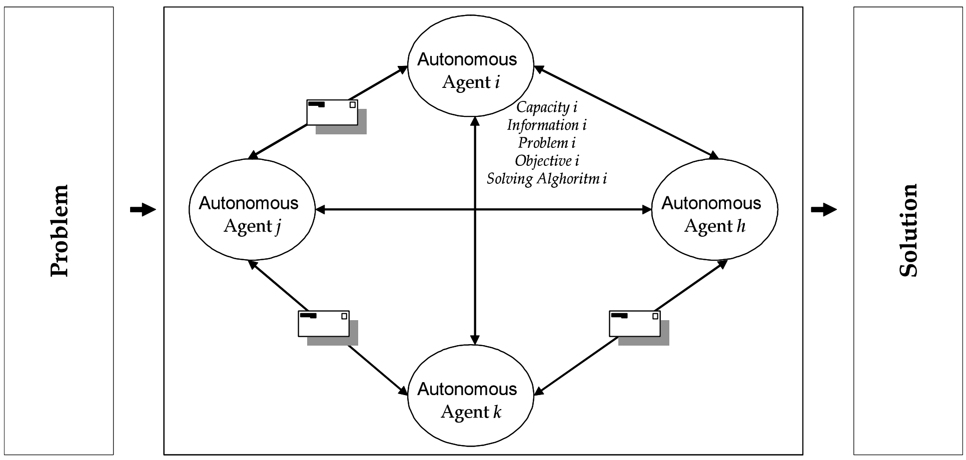


**Figure 3: A Multi Agent System**

The increasing development of the multi agent system are due mainly to their efficiency and robustness, to the ability to allow the cooperation of systems already existing and to the ability to resolve problems in which information, capabilities and control are distributed. Instead the MAS's main development difficulties regard the interaction and communication way among the agents.

The enterprise's managers take decisions being based on information and judgments coming from many environment. Ideally every important information should be made available before that decision is taken. However to obtain important and up to date information inside a big organization is a complex process and expensive in terms of time. For this reason, the organizations have decided to develop ICT – based systems in order to support the business process management. This problem can be faced representing a business process as a community of agents service's negotiators. The agents that demand a service from other agents start one negotiation for this service in order to obtain a price, a time and a degree of acceptable quality. The negotiation that conclude with successfully sanction an agreement among the parts. On this principle many applications of the agent's theory have been developed in various industrial sectors.

## 2.3 Agent Research and Developments

Before to discuss about the different research directions taken by agent technology it seems right to define what we mean by "agent", "agent-based system" and "multi-agent system". This is the first difficulty when talk Jenning, Sycara and Wooldridge (Jenning, Sycara and Wooldridge, 1998) in represent well the importance of agent technology which this statement:

*Autonomous agents and multi-agent systems represent a new way of analysing, designing, and implementing complex software systems.*

This statement brings in evidence the relation between agent systems and software systems. They define an agent in the following way:

*An agent is a computer system, situated in some environment that is capable of flexible autonomous action in order to meet its design objectives.*

For the scopes of this work the reference to computer systems can be too restrictive because we would like to use the agent paradigm to model and to simulate decision maker taking part to the supply chain related activities. We can indeed formulate an adapted definition of agent:

> *An agent is an actor, situated in some environment that is capable of flexible autonomous action in order to meet its design objectives.*

The term actor is the same intended in the UML notation.

There are thus three key concepts in our definition: *situatedness, autonomy,* and *flexibility*.

Situatedness, in this context, means that the agent receives sensory input from its environment

Autonomy can be intended in the sense that the system should be able to act without the direct intervention of humans (or other agents), and that it should have control over its own actions and internal state.

By flexible, we mean that the system is:

- *responsive*: agents should perceive their environment and respond in a timely fashion to changes that occur in it;
- *pro-active*: agents should not simply act in response to their environment, they should be able to exhibit opportunistic, goal-directed behaviour and take the initiative where appropriate;
- *social*: agents should be able to interact, when appropriate, with other artificial agents and humans in order to complete their own problem solving and to help others with their activities.

*Multi-agent systems* are ideally suited to representing problems that have multiple problem solving methods, multiple perspectives and/or multiple problem solving entities. Such systems have the traditional advantages of distributed and concurrent problem solving, but have the additional advantage of sophisticated patterns of interactions. Examples of common types of interactions include: cooperation (working together towards a common aim); coordination (organising problem solving activity so that harmful interactions are avoided or beneficial interactions are exploited); and negotiation (coming to an agreement which is acceptable to all the parties involved). It is the flexibility and high-level nature of these interactions which distinguishes multi-agent systems from other forms of architectures and which provides the underlying power of the paradigm.

Traditionally, research into systems composed of multiple agents was carried out as a field of Distributed Artificial Intelligence (DAI), and has historically been divided into two main camps: Distributed Problem Solving (DPS) and Multi-Agent Systems (MAS).

More recently, the term “multi-agent systems” has come to have a more general meaning, and is now used to refer to all types of systems composed of multiple (semi-)autonomous components.

Distributed problem solving (DPS) considers how a particular problem can be solved by a number of modules (nodes), which cooperate in dividing and sharing knowledge about the problem and its evolving solutions. In a pure DPS system, all interaction strategies are incorporated as an integral part of the system. In contrast, research in MAS is concerned with the behaviour of a collection of possibly pre-existing autonomous agents aiming at solving a given problem.

> *A MAS can be defined as a loosely coupled network of problem solvers that work together to solve problems that are beyond the individual capabilities or knowledge of each problem solver. These problem solvers agents are autonomous and may be heterogeneous in nature.*

The characteristics of MAS are:

- each agent has incomplete information, or capabilities for solving the problem, thus each agent has a limited viewpoint;
- there is no global system control;
- data is decentralized;
- computation is asynchronous.

Some reasons for the increasing interest in MAS research include: the ability to provide robustness and efficiency; the ability to allow inter-operation of existing legacy systems; and the ability to solve problems in which data, expertise, or control is distributed. Although MAS provide many potential advantages, they also face many difficult challenges.

One of the first models of multi agent-problem solving was the actor's model (L. Adacher, A. Agnetis, and C. Meloni, 2000). Actors were proposed as universal primitives of concurrent computation. Actors are self-contained, interactive autonomous components of a computing system that communicate by asynchronous message passing. The basic actor primitives are:

- create: creating an actor from a behaviour description and a set of parameters, possibly including existing actors;
- send: sending a message to an actor;
- Become: changing an actor's local state.

Actor models are a natural basis for many kinds of concurrent computation. The low-level granularity of actors also poses issues relating to the composition of actor behaviours in larger communities, and achievement of higher level performance goals with only local knowledge.

Another important building block for multi-agent research was the studies on Task Allocation through the Contract Net Protocol. The issue of flexible allocation of tasks to multiple problem solvers (nodes) received attention early on in the history of DAI (Davis R, and Smith RG, 1983).Davis and Smith's work resulted in the well-known Contract Net Protocol. In this protocol, agents can dynamically take two roles: manager or contractor. Given a task to perform, an agent first determines whether it can break it into subtasks that could be performed concurrently. It employs the Contract Net Protocol to announce the tasks that could be transferred, and requests bids from nodes that could perform any of these tasks. A node that receives a task announcement replies with a bid for that task, indicating how well it thinks it can perform the task. The contractor collects the bids and awards the task to the best bidder. Although the Contract Net was considered by Smith and Davis (as well as many subsequent DAI researchers) to be a negotiation technique, it is really a coordination method for task allocation. The protocol enables dynamic task allocation, allows agents to bid for multiple tasks at a time, and provides natural load balancing (busy agents need not bid). Its limitations are that it does not detect or resolve conflicts, the manager does not inform nodes whose bids have been refused, agents cannot refuse bids, there is no pre-emption in task execution (time critical tasks may not be attended to), and it is communication intensive.

# 3

# THE REFERENCE MODEL

## 3.1 Introduction

During last years, a lot of studies done in field of complex system management have shown how modelling is one of the key activities in understanding, designing, implementing, and operating systems (Camarinha-Matos *et all*, 2006). A model lets to predict system behaviour, 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 modelling 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 work I 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 guidelines able to support the designer in the system modelling phase of MAS development while indicating in which scenario is more convenient to adopt an approach over another one. The remainder of this chapter presents a reference model with a short discussion about model presented in the literature for MAS building.

## 3.2 Methods for multi-agent model development

The concept of Multi-Agent System was firstly introduced in computer science field by Jennings N.R., and M. Wooldridge in (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 modelling 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

My 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 modelling, statistics, and algorithms in order to make decisions in complex real-world problems. In this work I 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 I suggest the Unified Modelling Language (UML ) modelling which is able to represent the complex system with an object (agent) - oriented view: (Bauer et al, 2005).

## 3.3 The Reference Model

A Reference model in systems and software engineering is a model of something that embodies the basic goal or idea of something and can then be looked at as a reference for various purposes.

There are a number of concepts rolled up into that of a 'reference model.' Each of these concepts is important:

- Abstract: a reference model is abstract. The things described by a reference model are not actual things, but an abstract representation of things. Therefore, when describing the architecture of a house, an actual exterior wall may have dimensions and materials, but the concept of a wall is part of the reference model. One must understand the concept of a wall in order to build a house that has walls.
    - *Entities and Relationships*: A reference model contains both entities (things that exist) and relationships (how they interact with one another). A list of entities, by itself, is not sufficient to describe a reference model.

- *Within an environment*: A reference model does not attempt to describe "all things." A reference model is used to clarify "things within an environment" or a problem space. To be useful, a reference model should include a clear description of the problem that it solves, and the concerns of the stakeholders who need to see the problem get solved.
- *Technology Agnostic*: A reference model is not useful if it makes assumptions about the technology or platforms in place in a particular computing environment. A reference model is a mechanism for understanding the problems faced, not the solutions involved, and as such, must be independent of the selected solutions in order to provide value to the practitioner. Note: That does not preclude the development of a reference model that describes a set of software applications, because the problem space may be "how to manage a set of software applications."

### 3.3.1 A Reference Model uses

There are many uses for a reference model. One use is to create standards for both the objects that inhabit the model and their relationships to one another. By creating standards, the work of engineers and developers who need to create objects that behave according to the standard is made easier. Software can be written that meets a standard, and developers can copy that software to use it again, or build a software factory that generates that code. When done well, a standard can make use of design patterns that support key qualities of software, such as the ability to extend the software in an inexpensive way.

Another use of a reference model is to educate. Using a reference model, leaders in software development can help break down a large problem space into smaller problems that can be understood, tackled, and refined. Developers who are new to a particular set of problems can quickly learn what the different problems are, and can focus on the problems that they are being asked to solve, while trusting that other areas are well understood and rigorously constructed. The level of trust is important to allow software developers to efficiently focus on their work.

A third use of a reference model is to improve communication between people. A reference model breaks up a problem into entities, or "things that exist all by themselves." This is often an explicit recognition of concepts that many people already share, but when created in an explicit manner, a reference model is useful by defining how these concepts differ from, and relate to, one another. This improves communication between individuals involved in using these concepts.

A fourth use of a reference model is to create clear roles and responsibilities. By creating a model of entities and their relationships, an organization can dedicate specific individuals or teams, making them responsible for solving a problem that concerns a specific set of entities. For example, if a reference model describes a set of business measurements needed to create a balanced scorecard, then each measurement can be assigned to a specific business leader. That allows a senior manager to hold each of their team members responsible for producing high quality results.

A fifth use of a reference model is to allow the comparison of different things. By breaking up a problem space into basic concepts, a reference model can be used to examine two different solutions to that problem. In doing so, the component parts of a solution can be discussed in relation to one another. For example, if a reference model describes computer systems that help track contacts between a business and their customers (Customer Relationship Management), then a reference model can be used by a business to decide which of five different software products to purchase, based on their needs. A reference model, in this example, could be used to compare how well each of the candidate solutions can be configured to meet the needs of a particular business process.

## 3.4 The Multi Agent Reference Model

The reference model described in the **Figure 4** 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 4**) 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. Then, the simulation step allows the verification of the impact that agent's decisions have on the achievement of the global objective.

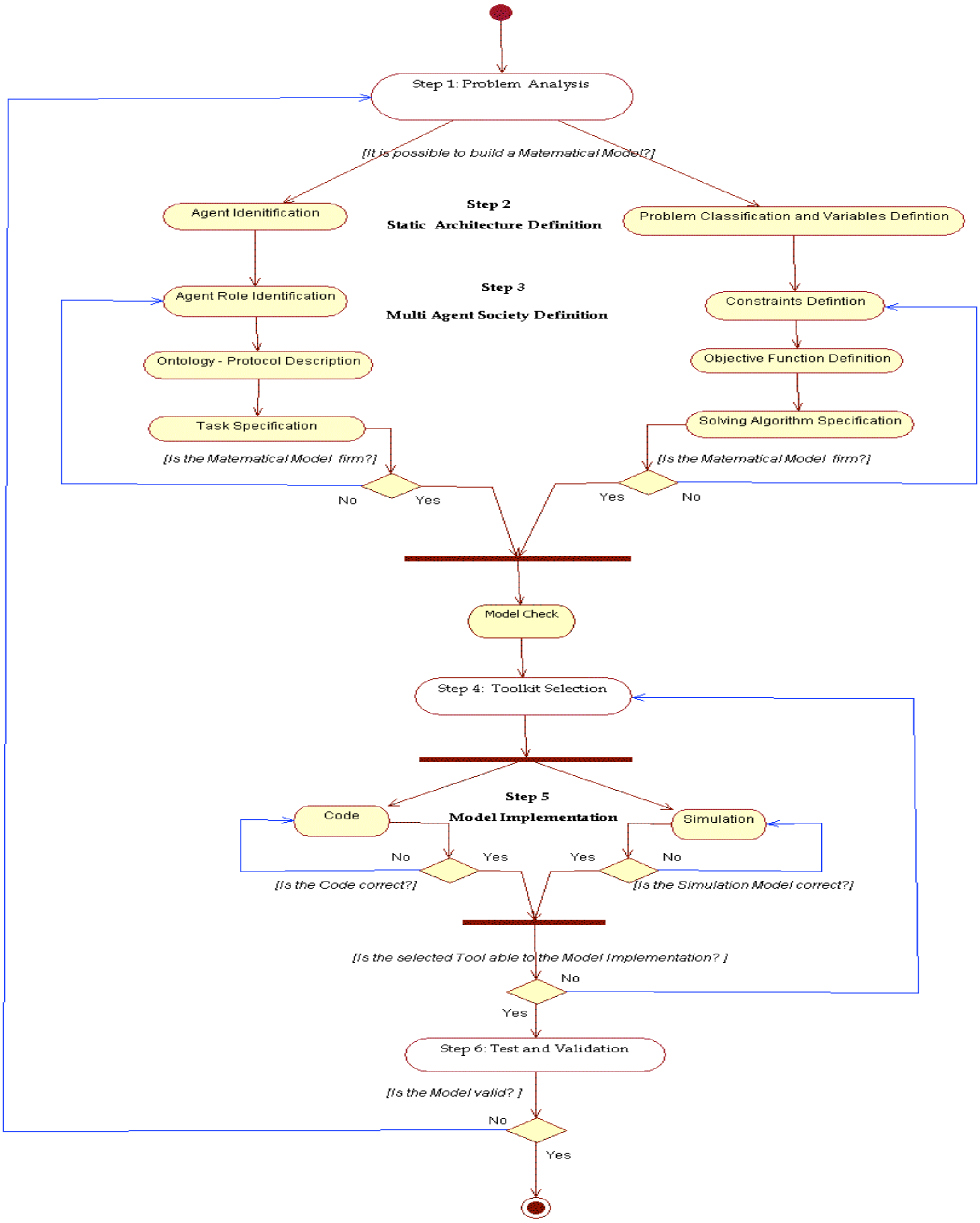


**Figure 4: The Reference Model**

The more important steps to build a Multi Agent Model are:

1. **Problem Analysis**: requirements analysis to represent a real system through a model.
    1.a. Domain Description: by means of UML (Unified Modelling Language) use cases diagram (i.e. manufacturing or logistic systems are described)
2. **Static Architecture Definition**: identification and description of real entities that will be the agents of system.
    2.a. Characterization Problem: by means of known models in literature,(i.e. the question is led back to scheduling or routing problem)
    2.b. Agents Identification: definition of agent's numerousness and type, (i.e. the agents can be declared functional or decisional)
    2.c. Variables Definition: definition of system's decisional parameters.
3. **Multi Agent Society Definition**: description of agent's role, activities and interactions.
    3.a.I. Role Identification: by means of UML sequence diagram it is possible describe the use cases in execution scenario.
    3.a.II. Tasks Specification: by means of UML activity diagram it is possible represent the processes that interest various agents.
    3.a.III.Ontology – Protocol Description: : by means of UML class diagram it is possible represent as a information is structured.
    3.b.I. Constraints Definition: identification of problem's constraints, in particular about the used resources, time available, supported costs..
    3.b.II. Objective Function Definition: declaration of problem and agent's objectives.
    3.b.III. Resolution Algorithm Definition: selection of most suitable algorithm able to resolve the Problem formulated to previous steps.
4. **Toolkit Selection**: this step regard the analysis and selection of the tools able to model the multi agent system designed in the previous steps.
5. **Model Implementation**: definition of complete multi agent architecture and of Agents behaviour.

5.a. Code: translation of multi agent system's fundamental concepts in chosen tool's programming language, (i.e. Java language if the chosen tool is Jade).

5.b. Simulation: the simulation process allows, to visualize the effect of some decisions on the system without that these must effectively be realized on the real process.

6. **Test and Validation**: this phase is relative to test and to validate the developed multi agent model created in term of architecture, tools and mathematical model. In this step, feedback system for modifications, or confirmations to the model is foreseen.

# 4

# REFERENCE MODEL APPLICATIONS

## 4.1 Production Scheduling in Innovative Flexible Manufacturing System

### 4.1.1 Introduction

The concept of Flexible Manufacturing System (FMS) was introduced in response to the need for greater responsiveness to changes in products, production technologies and markets and has be discussed deeply in literature (El Maraghy HA, 2006, Browne J et all, 1984, Felix T, 2004, De Toni A., Tonchia S., 1998). FMS have high degree of complexity and often are underused mostly due to lack in software systems and communication technologies able to overcome this hurdle. For this reason it is common to analyze the FMS along two different dimensions: the flexibility end complexity. The first can be analyzed as *internal flexibility*, as the ability to manage in efficient way the plant, and *external flexibility*, as the ability to quickly respond to the market requests. The latter (the complexity) is instead measured in terms of *(i)* plant complexity and, (ii) information domain complexity. The first is a indicator of the number of machines, of products, and of product models (Sarkis J, 1997). The second one is a function of total quantity of information, information diversity, and of information content, corresponding to the effort to capture and to traduce in useful format the information (El Maraghy WH and Urbanic RJ, 2003). The aim of this work is to present an application of a Multi-Agent System (MAS) able to solve in a short computing time a scheduling production problem with a high level of flexibility and complexity. In particular I present and confront a centralized and decentralized approach to solve the previous problem with growing level of information domain. The work represents a breakthrough step further with respect to the work (I.Baffo, G.Confessore, G.Stecca, 2007) and it is arranged as following. In the section 4.1.2 I present a discussion about the several approaches of modelling a complex system as a FMS. In the section 4.1.3 I introduce the case study and demonstrate the advantages of use a multi-agent model in order to not increase the algorithmic complexity of the model. In the Section 4.1.4 an application to a shoe manufacturing plant is given. Finally, in the last section of this paragraph concluding remarks are discussed

### 4.1.2 FMS as a complex system

In my work I refer to FMS as a complex system in which the interaction among the elements of the system, and the type of interaction between the system and the environment, do not enable the nature of the system to be understood simply by analyzing its components (Cilliers P., 1998) . For these reasons a discussion about the best efficient approach to model a behaviour of components and of system with a high level of complexity it is open. Often the Petri nets, offering a rigorous and analytic method, are preferable for modelling the control structure of distributed systems. However, the use and application of this model is limited by two problems: (i) the Petri net model become too large and complex even for a modest size problem (Wang J, Deng Y, 1999); (ii) the difficulty to make even a little change to a previously built model. The MAS approach aim to overcome these limits of rigidity and computing complexity in the FMS modelling (S. Fujii, T. Kaihara and H. Morita, 2000). The advantages offered by the MAS in this contest are: (i) the possibility to solve complex problem solving a set of easier local problems; (ii) the opportunity to change some problem parameters or to substitute any system element without discard entirely the original model. The difficulty of apply this model approach reside into the activity of architecture design used by the system elements to speak and act. The number and the kind of relations among the system entities became the most important indicator of the complexity model. In this work I want demonstrate that, applying a MAS approach to the same architecture, an increase of information quantity:

- not increase the computing complexity of the problem;
- increase the internal and external flexibility of the plant;
- increase the proximity of the model respect to real system.

### 4.1.3 A Multi Agent System Model For scheduling problem in FMS

We consider an agile shoe manufacturing plant with innovative transportation line. The innovative molecular structure of the transportation allows the products to overtake along production lines, increasing the overall flexibility of the plant (Martinez Lastra JL, Colombo AW, 2006). The basic element of the molecular structure is the “Tern”, which is constituted by two rotating tables, called “Table” and “Island”, and by a rotating three arms manipulator. The Terns move the parts to be worked while the workmanships are performed by the machines collocated around the Islands. An initial warehouse inserts the forms in the system, and a final storehouse picks up the produced shoes. The Table is used to direct the semi-finished shoes either to the next Tern or to the Island of the same Tern. Each table houses *twelve slots* on which the semi-finished shoes can be placed. Moreover, it moves backward the lasts flowing back towards the warehouse (the last is the object around which the semi-finished shoe is built upon). The Island directs the semi-finished shoes towards the different machining stations, laid around the Island itself. Each island instead houses *twenty-four slots*.

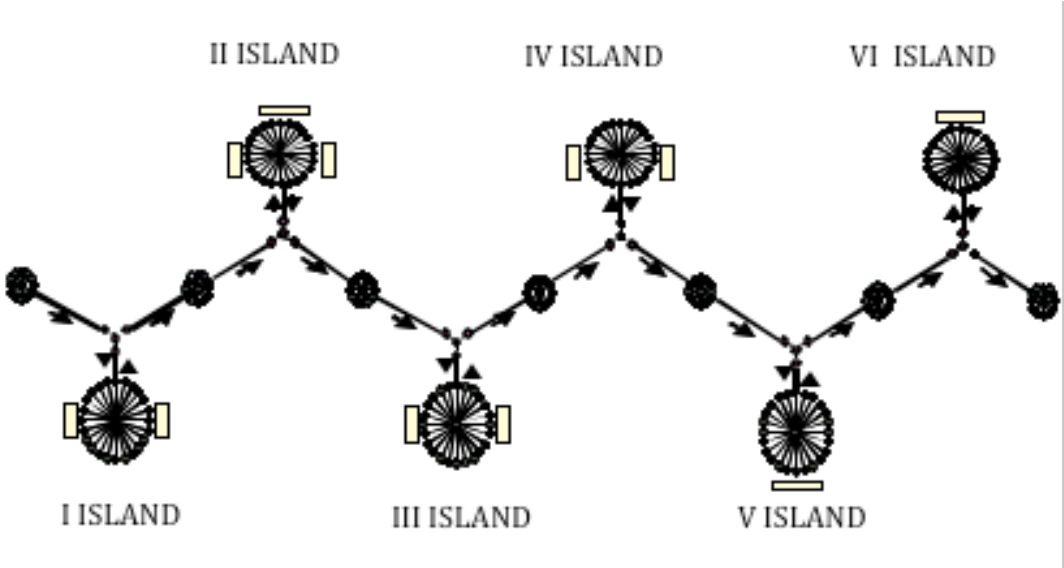


**Figure 5: Transportation line of the plant**

The processing time of the jobs on several island can be calculate as following:

- In a deterministic way with a $T_{det}$ formula;
- In a not deterministic way using a average between the case with the isle full of jobs and the case with the isle empty.

In a determinist way we use a Tdet for calculate the processing time of the jobs on the isles:

$T_{det}$ = [N(j-k) + N(i-k)] * max {t(1), t(2)} + [N(i) + N(j) - (N(j-k) + N(i-k))] * t(1) + [N(i) + N(j) + N(k) - (N(j-k) + N(i-k))] * t(2) + + [(N(r) + (N(i) - 1)) - [N(j-k) + N(i-k)] - [N(i) + N(j) - (N(j-k) + N(i-k))] - [N(i) + N(j) + N(k) - (N(j-k) + N(i-k))] * t(s) *(1)*

Where:

- N(r) = 24 is the total number of island's slots;
- N(i) is the set of jobs to sequence;
- N(j) is the set of jobs that are on the island before that the first machine;
- N(k) is the set of jobs that are on the island between the machine 1 and 2;
- N(i-j) is the set of couples belonging to set N(i) and N(k) and distant among them 8 slots;

- N(j-k) is the set of couples belonging to set N(j) and N(k) and distant among them 8 slots;
- T(1) is the job processing time on the machine 1. It is the same for every job;
- T(2) is the job processing time on the machine 2. It is the same for every job.

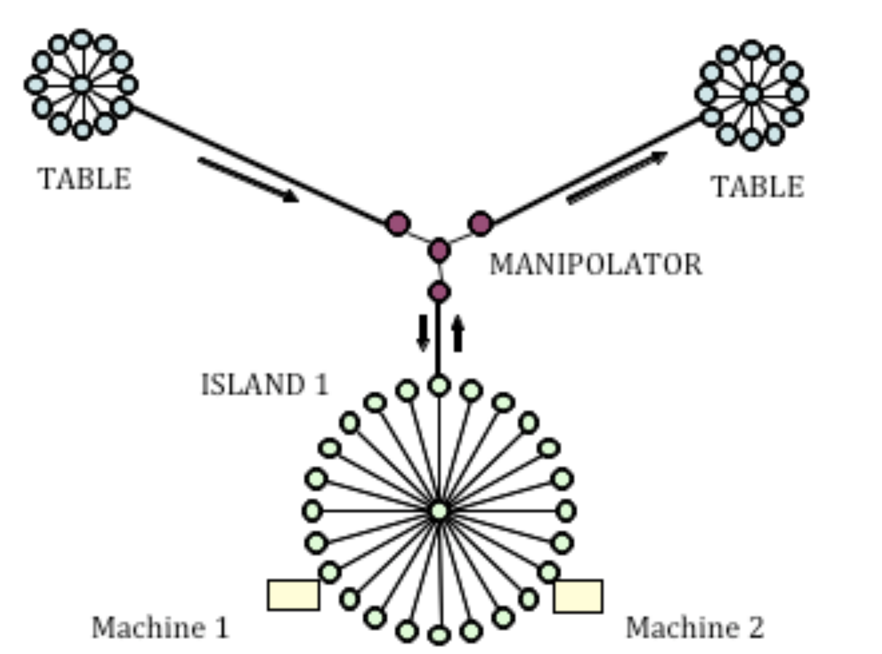


**Figure 6: Island - Cell Manufacturing**

The formula *(1)* is applied to every island of the plant and the total processing time is calculated both in approximated or deterministic way.

### 4.1.4 The Centralized Approach

In the centralized model the enterprise information system acts as the Coordinator Agent (CA) having total decisional and communication power. The CA receives from the market the orders consisting of the specification of products to be produced and the quantity, its due dates, and the release dates of the jobs. The CA has a complete information about the plant characteristics, such as the all parts processing time, computed in approximated or deterministic way. Using this information it can solve the scheduling problem and communicate the starting times at the first island. The best sequence of jobs to be processed is chosen using the EDD (earliest due date) rule. The rule sequences the jobs in not - decreasing order of their due dates and gives an optimal sequence for the single machine maximum lateness problem (1||Lmax). This scenario do not exploit the overtake capability of the system: the flexibility offered by the transportation line is nullified by the decision system rigidity.

### 4.1.5 The Decentralized Approach

In this model there are different intelligent agents, each one with an incomplete information about the plant and environment configuration. The agents are connected through a local network and they can communicate and cooperate in order to reach a shared solution. Therefore the negotiation is not necessary because the agents are not competitors. They are cooperative actors that share local information to estimate the goodness of different solutions. There are two types of agents: The Coordinator Agent (CA) who represents the Information System and the Island Agents (IA) who represent the manufacturing islands. CA has coordination role and it decides if to make changes to the already released production plan based on changes in market request in terms of due dates. For taking this decision it minimize the following decisional function:

$$f = \alpha * \Delta T + \beta * \Delta Cmax \qquad (2)$$

where α is a parameter who depends on order priority and β is a parameter depending on how many jobs must be processed, ΔT is the change in tardiness of the jobs and ΔCmax is the change in maximum completion time after schedule change. Each IA can send or receive messages only to coordinator agent or to neighbour islands. It decide the sequence of jobs to process solving the 1|r|Lmax problem, who consider release dates and it is solved by the agent using Branch & Bound. Agents send and receive messages to share the information and to obtain one shared solution using the overtake coordination protocol (OCP).

The OCP consist of the following steps:

1. The CA receives orders from the market and communicate at the first IA the release and due date of the jobs.

2. The IA solve a local scheduling problem and communicate to next Island Agent when it can start to process the jobs.
3. If due date of one job change during production, then the CA communicate the new due date to the Island Agent following to Island Agent who is working the changed job.
4. The CA find the best schedule and send to CA this schedule with ending times for every job.
5. The CA calculates the convenience of changing schedule using the function *(2)* and, if overtake is convenient, sends overtake order to the IA.
6. The last IA process the jobs and sends the products to finished product warehouse.

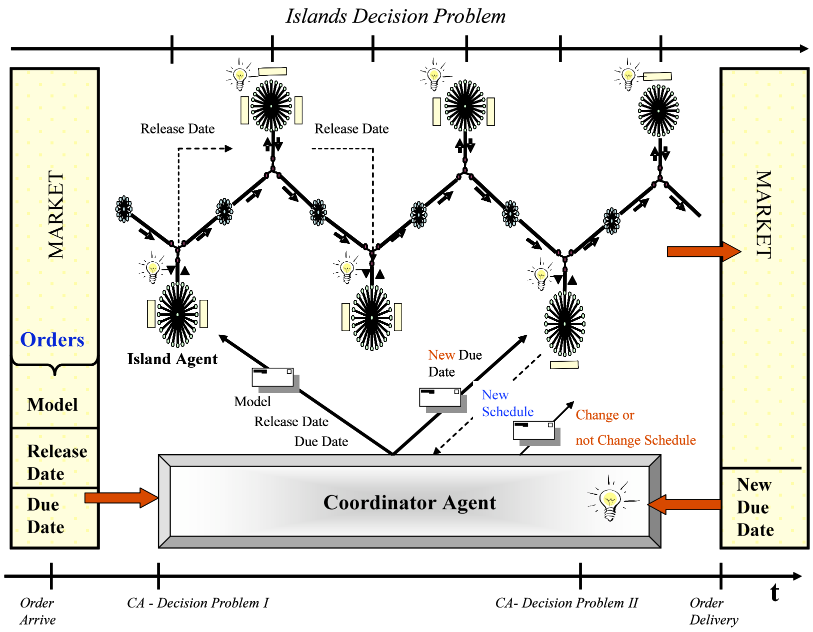


**Figure 7: Communication Protocol**

### 4.1.6 Test Results

I implemented a simulation model using Java programming language. We present two exemplificative scenarios of job due date change after the production plan is released to the plant, showing the different decision strategy of the Coordinator Agent (CA) using the deterministic and approximate approach. In the first scenario the due date of job 3 change after the first island and the processing time is calculate as average of these two case: island full and empty of jobs. In the second scenario the change of the due date job is the same, but the processing time is deterministic and obtain applying to every island the formula (1).

**Table 1: Due date of three jobs at t=0 and**

| Job | dd (t=0) | dd' (t=1) |
|---|---|---|
| 1 | 300 | 300 |
| 2 | 350 | 350 |
| 3 | 400 | 300 |

The **Table 1** displays the job information in the two scenarios. In particular, job number, due date before and after the change are reported.

**Table 2: Model simulation**

| No Det Approach | t | Seq | Cmax | Delay | f | S |
|---|---|---|---|---|---|---|
| Centralized before dd change | 0 | 1,2,3 | 594 | 194 | | |
| Centralized with new dd | 1 | 1,2,3 | 594 | 294 | 888 | |
| Decentralized with new dd | 1 | 3,1,2 | 583 | 283 | 866 | X |
| **Det Approach** | **t** | **Seq** | **Cmax** | **Delay** | **f** | S |
| Centralized before dd change | 0 | 1,2,3 | 587 | 187 | | |
| Centralized with new dd | 1 | 1,2,3 | 587 | 287 | 874 | X |
| Decentralized with new dd | 1 | 1,3,2 | 651 | 351 | 1002 | |

The **Table 2** displays the solution found by the centralized and decentralized models for the two scenarios, and before and after the due date change (denoted with time t = 0 and time t = 1 respectively). In particular the sequence, the maximum makespan (Cmax) and the delay of the job with due date changes are reported. In the last column of the table a *X* mark the best solution of the scenario. In the first scenario, with deterministic time, if we assume that α is the same of β then the Coordinator Agent is willing to change the schedule because the worsening of the *Cmax* is smaller then advantages of reducing the tardiness of job 3. In this case the *external* *flexibility* meant as the ability to respond quickly to market change is more important then completion of all jobs early. In the second scenario, with approximated processing time, if we assume α equal to β, the Coordinator Agent is willing to do not change the schedule, because the advantage of reducing the tardiness of job 3 do not compensate the worsening of the *Cmax*. In this case, the *internal flexibility*, meant as the ability to manage in efficient way the plant is more important that to respond quickly to market requests. In both cases the value of α and β determine the decision of CA. This theoretic example show how the introduction of an higher level of information in the system not increase the computing complexity and it can determine different decision about the plant. In particular, in the first scenario, the CA chose a decentralized approach but with approximated information about the job's processing time. With the deterministic time the decision change and the advantages of the model are major using a centralized approach. In general, the use of determinist information allow to avoid mistakes about the value of job's processing time estimate. As reported in **Figure 8** and **Figure 9**, the errors of this approximation are: *(i)* 8% respect to deterministic time, using a centralized approach; *(ii)* 5% respect to deterministic time adopting a decentralized approach.

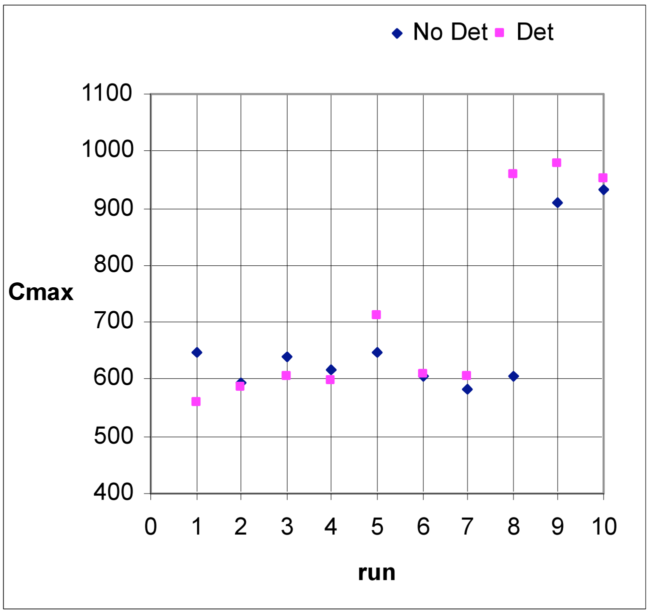


**Figure 8: Comparison between deterministic and not deterministic job's processing times using a Centralized Approach**

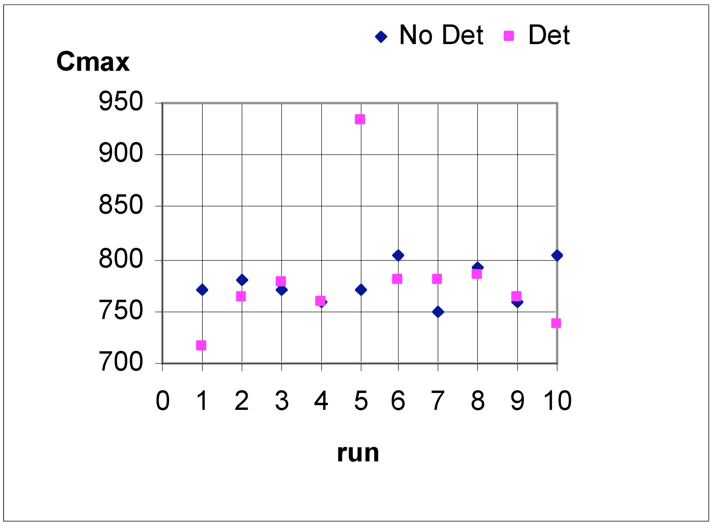


**Figure 9: Comparison between deterministic and not deterministic job's processing times using a Decentralized Approach**

### 4.1.7 Conclusion

In this application a multi-agent model able to model complex systems like FMS are presented. The advantages offered by this approach are demonstrated by an application to a shoe manufacturing plant. The results of the system simulation implemented with java language demonstrated that:

- a decentralized approach can increase the flexibility of the plant allowing the job overtakes when this solution is considered optimal by the Coordination Agent.
- an increase of plant's quantity information in the model not require an increase of complexity in the system and in particular in the island agents.
- the introduction on the model of deterministic information and of a decisional function, allow at the same time to have an instrument more flexible and closer to the real system.

## 4.2 Application to Lean Healthcare System

### 4.2.1 Introduction

The cost of services provided by public and private healthcare systems is nowadays becoming critical. Since in several countries, hospital revenues are not more cost based, reduction in costs are becoming the critical point for the sustainability of all medical structures. Moreover, the great size of the hospital, while on one hand allows to exploit the economies of scale, on the other hand makes the internal dynamics of a hospital a complex non-linear structure (Harper 2002) that is difficult to manage.

This work tackles the issues of hospital equipments, personnel and drugs management by emphasizing its implications on the whole healthcare system efficiency. The adopted approach is based on the assumption that the hospital is a complex system composed by a great number of entities and processes. Camarinha-Matos and Afsarmanesh (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 of a Multi Agent System (MAS) to collaborative network modelling turns out to be natural.

The MASs have been widely exploited in several application fields in order to govern the complexity through cooperation and decentralization of decisions when competitive requests and divergent objectives have to considered. In this work a MAS-based approach is adopted in order to offer a model able to manage the complexity of a healthcare system while:

- fostering technological innovations through information sharing and recovery;
- fostering organizational innovations for more efficient and effective distributed decisions whenever possible.

A MAS can be herein considered as the basic methodological factor for then deploying technological enablers (e.g., ICT) and managerial enablers (e.g., a cooperative decision system of the healthcare organization). The system is supposed to consist of several decisional and operational agents that can be related to departments or wards. Every agent is considered as a single decisional entity that influences the achievement of system's goals. The work presents a model of agents' cooperation for addressing the problem of departments, wards and personnel integration in order to improve the efficiently regarding drugs, equipments, and personnel management. The overall goal of the system is related to the need of providing, with a fixed budget and consequently a fixed set of resources, the highest level of customers served with high service level The work presents a model of agents' cooperation for addressing the problem of departments, wards and personnel integration in order to improve drugs, equipments, and personnel management. In particular, the cooperative model for drugs management is based on cooperative actions for obtaining service effectiveness while reducing inventory costs through information sharing and coordination. According to Camarinha-Matos and Afsarmanesh (2007) more than one modelling perspective must be integrated in order to properly design a Collaborative Networked Organization (CNO). I will describe architectural aspects presenting the agents and their roles in the CNO, component aspects describing technological aspects in terms of hardware/software needed in the CNO in order to fulfil the overall objective, functional aspects describing peculiar processes of the CNO.

The paper is organized as follows: Section 4.2.2 presents the main issues concerning healthcare management systems with respect to the management of equipments and drugs. The problem is described in Section 4.2.3. In Section 4.2.4 the agent-based cooperative model is presented and discussed. Conclusions follow.

### 4.2.2 Resource management in healthcare systems

The efficiency of healthcare systems is influenced by increased demand for quality, technology investments and increased drug supplies (Athanassopoulos and Gounaris, 2001). The consequent trade-offs among effectiveness, efficiency and equity objectives have fostered the development of theory and applications concerning health economics and management science (Athanassopoulos and Gounaris, 2001).

Since in several countries, hospital revenues are not any more cost based, cost reductions are becoming the critical point for the sustainability of all medical structures. For this reason some managers tried to apply some principles and techniques deriving from management science, for improving the efficiency and effectiveness of hospital facilities. In order to obtain cost savings, hospitals need to review their activities, to identify the costs associated with the activities and reduce them, to classify the activities in terms of added-value, and to decrease or eliminate not added-value activities (Aptel and Pourjalali 2005). By applying this methodology, called Activity Based Management (ABM), a lot of organizations focused their efforts to improve activities belonging to the logistic department. This department is a vital part of a hospital because it may have responsibilities for activities like purchasing, receiving, inventory management, management information systems, telemedicine, transportation, and home care services. Although these activities do not represent the core mission for the hospital, they take part into definition of service level offered by the organization. Consequently, it is important to examine the activities of this department to improve services and possibly reduce costs while adopting lean service processes. The internal dynamics of a hospital represents a complex non-linear structure (Harper 2002). Planning and management of hospital daily operations require a thorough understanding of the system with information for decision making (Harper 2002).

Literature review confirms that several studies address management issues although many gaps need to be bridged, for instance with respect to integrated logistics of healthcare systems. De Angelis et al. (2003) investigate the problem of assigning resources and servers (e.g., doctors, beds, instruments) to services. Akcali et al. (2006) tackles the problem of optimizing hospital bed capacity planning through a network flow approach. Harper (2002) proposes an integrated simulation tool (PROMPT, Patient and Resource Operational Management Planning Tool) for the planning and management of hospital resources such as beds, operating theatres, and needs for nurses, doctors and anaesthetists. Van Merode et al. (2004) study the potential adoption of Enterprise Resource Planning (ERP) systems in hospitals while facing the issues of planning and control processes and determining ERP systems requirements.

Liu et al. (2001) and De Treville et al. (2006) argue that efficiency and quality of healthcare management have been enhanced by the computerization of healthcare information. Several studies concern the information management, workflow (see Dang 2008) and automation in healthcare systems. Thornett (2001) discusses potential roles, introduction benefits and difficulties related to computer decision support systems within the practice in primary healthcare. It can be concluded that organizational innovations and enabling ICT-based solutions are essential conditions to reach the efficiency and effectiveness improvement in hospital operations management.

### 4.2.3 Problem Description

Thanks to the effort of an Italian research project's partners it was possible to analyse and synthesize the main problems related to drugs and equipment management within some Italian hospital structures. In particular, the study of the AS-IS processes and the survey on the employers showed several problems in the following processes: (*i*) central drugs management, (*ii*) drugs management in each ward, (*iii*) medical equipment management.

With regard to the first aspect, the problems arise from some lacks in the continuous monitoring of the central drugstore. When the responsible of the drugs is not present, the nurses, filling a paper register, are free to use the pharmacy. That approach could produce an inappropriate use of the drugs or errors in the related information records. Other problems come from the variability of the drugs in terms of codes and packaging. When a drug order is fulfilled, it is possible that the required drug it is not available. Consequently, it is substituted with an equivalent drug. It was observed that equivalent drugs are registered with different codes. This approach generates difficulties in the drugstore management. Regarding the drugs management in each ward, instead, the followings problems can describe the scenario analyzed by researchers, medical personnel and industrial managers:

The stock levels of the wards drugstores can be out of control whether not recorded in the computer system of the hospital.

The drug order fulfilment of the central drugstore could not properly take into account the stock levels of wards drugstores. Since information on the availability of medicines in the wards drugstores could not be recorded into the computer system, the nurses may need to check the availability by physical inspection in the wards.

The decisions about the fulfilling of drugs are made by the nurses. Minimum supply concept is not used in the fulfilling process. The orders are often made without information about the stock availability. That approach can generate an over abundance of drugs. Furthermore, it was observed that the nurses tend to make large orders in order to increase their responsiveness capability.

Problems arise in the management of medicinal products expiration dates with possible unreasonable costs of not usable drugs. With regard to the movements of medicinal products between the wards drugstores, it can happen that the nurses of a ward instead of replenishing the local drugstore by using the central pharmacy of the hospital, take the medicinal products from other wards drugstores. This procedure can imply that the nurse do not physically move throughout proper the hospital structure and consequently the drugs that have been taken could not be registered. It was observed that the movements of medicinal products can tend to generate conflicts between nurses of different wards.

Other problems coming from medical equipments management are:

- High cost of the management of hospital equipments. It has to be considered that the weekly movement of the required materials take 1.5 workdays of the store employee.
- The hospital equipments store is not managed by a computer system. The stock level is not exactly known. In order to find the equipment necessary for a patient it is often necessary for a nurse to move physically into the store where there is not a standard procedure to find the selected equipment but it needs to be founded by visual inspection.

- The localization of the equipments into the hospital structure is not defined. When something is not found in the equipments store, the nurse needs to check this stuff in all the hospital building.
- Tags are used to assign each instrument to each patient. The tag can be lost or it deteriorates too much, making it ineffective with the risk of loosing data of the patients.

Currently, several internal possible solutions are object of studies to improve the quality of the drug management. It seems that the most efficient manner to organize the system is to create a centralized unit, represented by the central drugstore, with higher control on the whole drugs and equipments management process.

### 4.2.4 Methodology: the cooperative model

This section proposes a model based on multi-agent theory in order to formalize the cooperative model under two dimensions: (*i*) organizational innovation for cooperative process and (*ii*) technological framework supporting the cooperative process. Moreover, as Camarinha-Matos and Afsarmanesh (2007) claim, the collaborative network requires the development of models, not only as a help to better understand the area, but also as the basis for the development of methods and tools for better decision-making (Camarinha-Matos and Afsarmanesh 2007). The realized model will be illustrated through the description of the agents, the explanation of the collaborative decisional process and the technological architecture. Finally, a discussion on ongoing and foreseen applications is made.

*Agent's Description*

The reengineered system should enable the collaboration among medical wards in the processes of material procurement and resource allocations. As described in Section 4.2.3, the lack of integrated information systems and the not formalized collaboration rules are the main obstacles to the development of a collaborative process. A model composed by a central manager and by medical ward managers is proposed. They act as interfaces of the central warehouse and the local ward warehouses. The *Department Manager* (DM) is a coordinator of resource allocation, and of material management processes. The *Ward Managers* (WM) represents the local decision maker for the ward and department objectives. The DM is a composition of three specialized decision makers: *(i)* Equipment manager, *(ii)* Drugs procurement manager, *(iii)* Human resource manager.

The DM is an enabler for centralized information and an enabler to collaboration among wards for drugs procurement, equipment allocation and human resource assignment.

*Collaborative decisional process among agents*

The decisional process is general and foresees an interaction between WMs and DM each time that a request is generated by the WM.

As showed in **Figure 10**, a WM sends a procurement request to the DM. The DM verifies the central and the local warehouses. Then, it sends a set of procurement options to the WM who replies with its choice. If the choice is an inter-ward delivery (for instance from WM B), then this request is forwarded to the ward chosen. The process terminates with a delivery from WM B to WM A and a notification of the delivery to the DM.

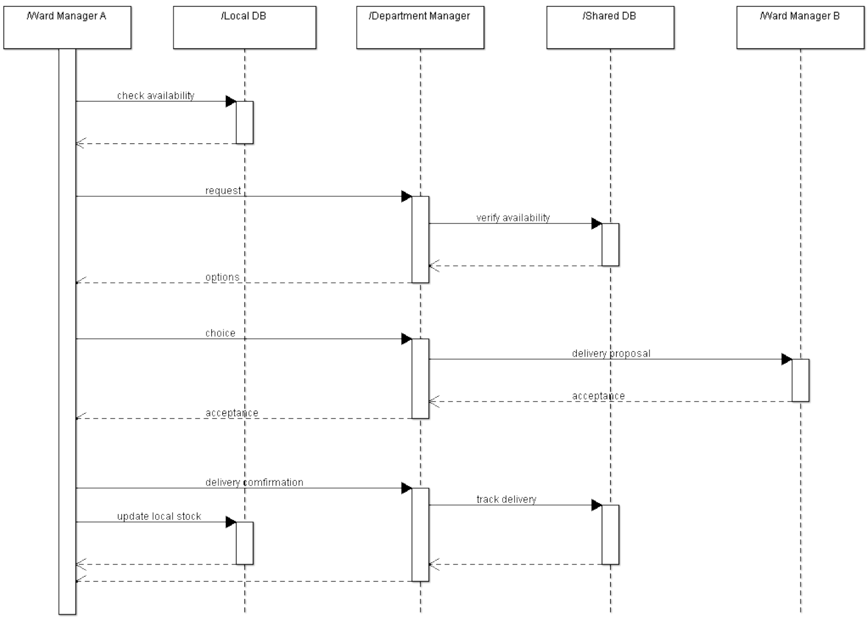


**Figure 10: UML sequence diagram for a general request**

The architecture shown in Figure 11 supports the collaborative process and the information centralization in a hospital system. The information systems must be fully integrated in order to track deliveries and material consumption.

Very promising technological enablers for effectively managing, even in real time, particular products or systems (equipment, medical devices, drugs) are the Radio Frequency Identification (RFId) technologies. By using RFId tags and devices it is possible to identify and to track a particular item inside or outside the medical structure. Moreover, it is possible to connect each item to an information system storing the history of usage and other logistic records. The data linked to the items can be used by decision technologies and tools based on proper cooperative models. The introduction in healthcare structures of a pervasive technology such as the RFId with complementary decision tools likely entails the redesign of key logistic processes as described in this work. On the other hand the presence of RFId technologies raises the complexity of the system. In fact, together with the introduction of RFId different critical issues must be resolved or managed such as privacy management and signal transmission interferences.

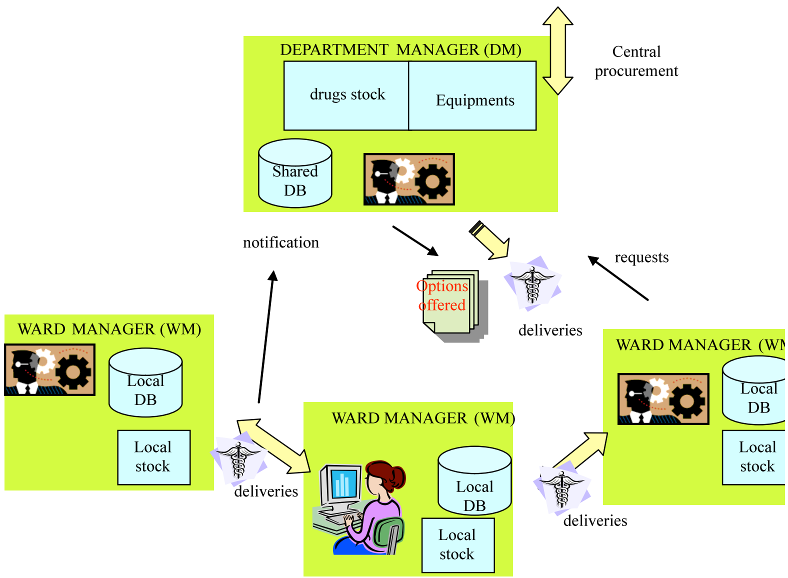


**Figure 11: Multi Agent System**

The tracking points for deliveries and material consumption are detailed in the collaborative procurement procedures. In order to enable physical tracking and localization, RFId technology can be used in combination with automatic delivery devices. Automatic delivery devices impose the authentication and enable the tracking of the delivery. Moreover, introducing the prescription of the doctors in the hospital information system, it is possible to perform a crosscheck between the prescribed drugs and the drug quantities that have been taken from the stores. Overall, three main areas of application of the system have been identified:

1. Drugs procurement;
2. Equipment management;
3. Human resource scheduling.

The application to drugs procurement allows the improvement of the FIFO material management strategy and effective use of decentralized warehouses. When the requests arrive to DM, the drugs manager verifies the stocks and propose different delivery options based on the expiry dates, distance from the warehouses, lead-time of procurement from suppliers. If an inter-ward delivery is chosen then it must be tracked. The ward from which the drug has been sourced will be supplied once the supplier procurement will arrive. The application to the equipment management foresees the localization and the tracking of the equipments uses. This information enables the correct allocation of the equipments. The requests of use of the equipment should be enriched with the information about the foreseen time of use and the scope of the use. The equipment manager tracks all the uses in order to plan maintenance and substitution with the objective of maintaining a high level of service in term of availability of each kind of equipments. The application to Human resource scheduling foresees the use of medical skills database owned by each human resource ward. The requests of wards should come in terms of skill requested and duration of the need. The Human resource manager should reply with the best available options. Each temporal transfer of medical personnel should be tracked in terms of its duration and of developed skill.

Through the adoption of the herein presented MAS-based approach for modelling the drugs and equipments management processes, it is expected in particular a better medicinal products inventory management while introducing process innovations and enabling technologies. The logistic department should then have a strategic role inside of a hospital due to the visibility on state and location of drugs, equipments, personnel, patients, etc.

### 4.2.5 Conclusion

In this paragraph a collaborative model for efficiency improvements in healthcare systems has been proposed. The proposed model faces the challenges of (*i*) gaining the benefits deriving from successful collaborative models already used in industrial systems and (*ii*) transferring the most appropriate industrial management practices to healthcare systems. The proposed approach could allow a better medicinal products inventory management. The drug expiration problems can be avoided and the costs significantly reduced. To develop the solution proposed in this paper the authors propose a MAS-based approach to model the drugs and equipments management processes. With this solution they propose an innovation of the processes based on re-design of operations and on the introduction of technological solutions. The logistic department acquires a strategic role inside of hospital because it can become the gathering point of all the hospital information related to state and location of drugs, equipments, personnel, patients, etc.

This work is partially supported by the Italian region "Lombardia" within the "Lean Healthcare" project. The project is going on and some of the solutions proposed in this paper are now considered for the testing phase within two Italian healthcare structures. The results of these tests with a comparison between the previous and the future performance in terms of time, costs and service level will be possibly presented in a progress of this paper. The presented work is based on part of the project' outcomes. The project joins industrial managers and researchers with the aim to transfer the most known industrial management practices to healthcare systems. The authors would like to acknowledge the whole Project' partnership, the

hospital structures that have allowed the analysis of their internal processes and the financing authority.

## 4.3 A Performance Indicators Model for integrated supply chain management

### 4.3.1 Introduction

In the last years, several studies done in the field of complex systems management such as supply chains, have shown how modelling is one of the key activities in understanding, designing, implementing, and operating systems (Camarinha-Matos & Afsarmanesh, 2006). A model let to predict systems behaviour, to evaluate different system problem solutions, to test different alternatives and to analyze their effect without doing changes on the real system. The huge potentialities offered by models can be exploited particularly in the representation of complex systems such as collaborative networks. Camarinha-Matos and Afsarmanesh in (Camarinha-Matos & Afsarmanesh, 2005) defined a collaborative network like 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 Systems (MAS) to collaborative supply chain modelling is natural. The MAS (Jennings & Wooldridge, 1995) are widely studied in literature as a method able to represent dynamic and distributed systems with several decisions makers having different information domains. This definition allows to translate the MAS concept into a more practice level, and to introduce the definition of Integrated Supply Chain (ISC). Indeed, an ISC can be defined as a network of suppliers, factories, warehouses, distribution centres and retailers that operate in integrated manner (Fox, 1992) with the aim of reducing inventory and costs, adding value, extending resource and accelerating the time to market of the extended Supply Chain. Usually in this kind of system the different actors are independent and execute one or more part of the whole supply chain process. In order to allow the achievement of a global objective, activities of coordination, negotiation and communication are necessary.

Moreover these activities must be supported by a flexible architecture able to face and promote the organizational innovation of the supply chain, key factor for a competitive advantage in the global market. For this reason, we propose a Performance Indicators Model (PIM) for the MAS creation like models of Integrated Supply Chains. In particular we propose an application of this PIM to a particular and realistic ISC with the following aims: (i) understand in which way every agent bring about value to the supply chain value; (ii) obtain an integrated indicator of whole supply chain performance,(iii) create a supporting framework to increase the performance visibility along the supply chain. From a system organization point of view this work allows to create a dynamic map of each agent interests, evaluating the opportunities or the hurdles proved by single chain's actor. Indeed, the model created for representing the supply network with the MAS approach, can be considered a useful tool for the strategic decision making process. The output of this tool application can be presented as a key indicator system of the Integrated Supply Chain (Otto & Kotzab, 2003). The challenge proposed by this work, is to evidence as in a integrated, collaborative and structured supply network, the final result, measured by the indicators system, is more efficient and present an added value higher then the sum of value offered by the single parts.

The remainder of this paragraph is structured as follows. Section 4.3.2 gives more details about the approach and presents a comparison with traditional model of measurement of the business process performance. In Section 4.3.3 the PIM is presented and argued. In Section 4.3.4 an application to a distributor of office solutions products is provided. Finally, conclusions and future works are discussed in Section 4.3.5.

### 4.3.2 Multi Agent System Approach to model a Business Process Measurement

The development of new technologies – and of the "organization and management science" - has conditioned the structure and the operations of production and delivery systems. The electronic and computer science, mechanical technologies and their integration have allowed the automation of many production processes and, contextually, the reduction of time to share information of some processes. Moreover, the new capabilities offered by ICT market, allow a more elevated degree of flexibility often not exploited due to lack of adequate organization systems. In this work I propose the MAS approach to fill this lack. In particular, I adopt a decentralized view in order to support the Supply Chain manager to aggregate the performance indicators for each agent that operate in the Supply Chain system.

The PIM proposed in this work it is thought to be not an alternative to most traditional models of business process performance measurement such as SCOR model (Supply-Chain Council, 2001) or Balanced Scorecard model (Kaplan & Norton 1996). The purpose of my model is indeed to adopt a decentralized view for measurement and analysis of a system's processes. I do not give any new guideline to improve the performance of a company's processes like the SCOR model. I offer instead an architecture for organizing a set of process's measures, assigning to the agents the responsibility of the performance corresponding to these measures. My work do not have the goal to propose a performance indicators set, it presents instead a model able to aggregate different existing measures for understanding in which way every agent bring value to the supply chain.

With the respect to balance scorecard I adopt an higher point of view. I share the importance to have indicators aligned with company's strategies, but I chose to measure a more wide system composed by customer, suppliers, carriers and warehouses.

Every actor of this chain exchange information and products with another, and acts according to the achievement of its goal, for these reason I call these actors "Agents". Every agent can influence, or better determine, the performance of the enterprise because the company is the customer's reference point. For this reason it is very important for the company understand which agent bring about value or product more inefficiencies for going to improve the performance with respect to own customers.

Following this aim I build a PIM that will be described in the next section.

### 4.3.3 Model Description

In this Section I describe a Performance Indicators Model, with the following aims:

(i) to understand in which way every agent bring about value to the supply chain value;

(ii) to obtain an integrated indicator of whole supply chain performance.

The construction of the model is performed in the following steps:

- analysis of the Integrated Supply chain;
- definition of the agents interacting in the integrated supply chain;
- definition of the information *tickets* and the *atomic indicators*;
- definition of the *agent indicators* as composition of atomic indicators;
- definition of the integrated supply chain indicator as composition of agent indicators.

The steps above listed can be schematized in **Errore. L'origine riferimento non è stata trovata.**. The starting points are the multi agent modelling approach and the process analysis. At this stage are defined the agents and the points in which the agents interacts with the integrated supply chain.

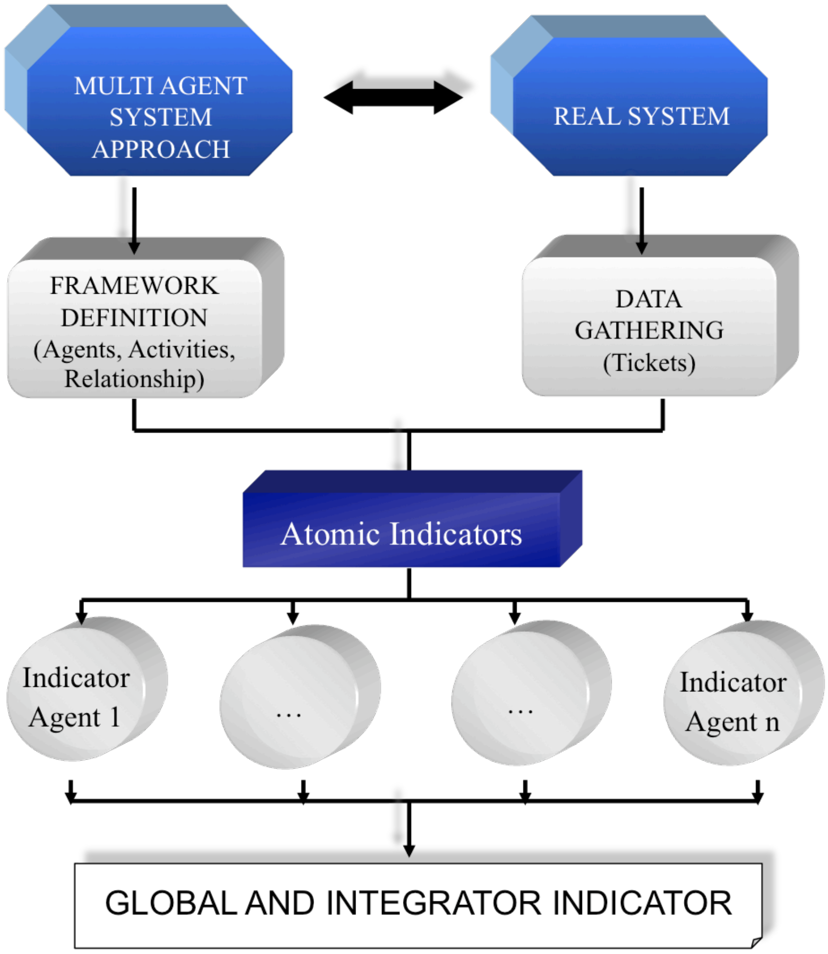


**Figure 12: Perfomance Indicator Model Scheme**

The interaction points are categorized in the phases of the supply chain (defined during the process analysis) and can be schematized as use cases (in UML notation) of the integrated supply chain system. In each phase a set of *tickets* are defined. A ticket is a source of information. An example of ticket is the number of order rows worked during picking process inside a warehouse. The ticket are used to build atomic indicators $A_k$, $k$ = 1, 2, .. $m$. An example of atomic indicator could be the number of order rows worked over the total number of order rows. The atomic indicators can be related to the agent who execute a process step. In this case I can denote atomic indicator as $A_{ijk}$ as the indicator $k$ of the agent $i$ in the phase $j$. At each phase $j$ of the global process, the performance of an agent $i$ can be measured with one or more indicators. For that stage the performance of the agent can indeed defined as the weighted sum of atomic indicators defined on the agent.

$$I_{ij} = \sum_{k=1}^{K_j} \alpha_{jk} I_{ijk} \qquad (3)$$

The performance of a supply chain phase can be defined as function of the performance of all agent interacting on that phases $IP_j = f(I_{1j}, I_{2j}, .., I_{Nj})$ while the performance of the agent $i$ can be defined as function of the performance of all phases in which agent $i$ interacts: $I_j = g(I_{i1}, I_{i2}, .., I_{iK})$.

Under the particular conditions (disjoint indicators) $IP_j$ can be written as product of the indicators of the agents interacting in the phase $j$:

$$IP_j = \prod_{i=1}^{N} I_{ij} \qquad (4)$$

Under the same conditions the performance of the agent $i$ can defined as the product of the performance of all phases in which agent $i$ interacts with:

$$I_i = \prod_{j=1}^{K} I_{ij} \qquad (5)$$

Having defined the performance of all agents interacting in the supply chain I can now define the performance of the overall supply chain as the product of all agent performance:

$$Isc = \prod_{i=1}^{N} I_i \qquad (6)$$

I can also define the effectiveness of the supply chain looking to the “goodness” of the agents interacting into the supply chain. If I could substitute the agents with the best performing on the market I will obtain a optimal performance defined as $I^{*}sc$. I will have a effectiveness factor defined as:

$$e = \frac{Isc}{I^{*}sc} \qquad (7)$$

This kind of indicators allows to quantify a benchmark analysis in the case of the market competitors adopt the same method of measurement of the supply chain performance with respect to customer satisfaction.

### 4.3.4 Model Application

The model application proposed in this paper refers to a company that develops and distributes, products, services and solutions for the business offices. The aim of this real application is to show as a logical and embedded framework that organize different indicators can help a company manager of supply chain to:

1. understand the operate of single agent with respect to company and customer's expects.
2. give more consistence to the strategy defining a target for the indicators that are aligned with the specific strategic objective.

The system analyzed in this work is related to Company, Carriers, Warehouse and Customer's performance. The Supplier's results don't belong to this study due to scanty data availability for the supplying processes.

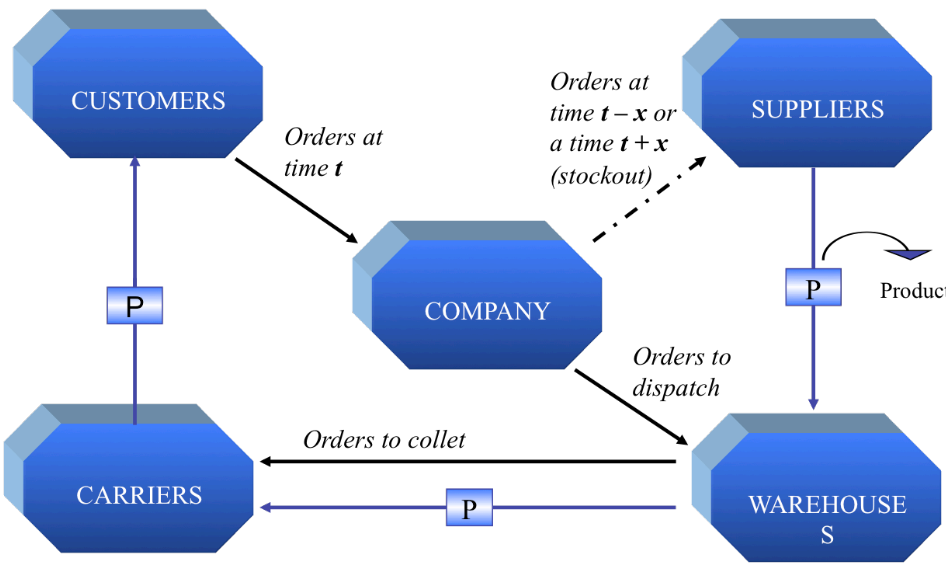


**Figure 13: The system analyzed**

*Framework Definition – Data Gathering*

The definition of the framework require the identification of the chain's agents and of the communication protocol among them. In this case, in particular, the process of customer orders dispatching can be described as following:

1. The Customer Agent (CA) sends an order of products or services to the company by fax, by commercial agent or by on-line platform.
2. The Company Agent (COA) receives the customer order and sends a Dispatch Order to the Warehouse manager.
3. The Warehouse Agent (WA) organize the shipment and communicates an order to collect to the carrier.
4. The Carrier Agent (CAA) is responsible to deliver the products to customer in the time agreed.

This kind of protocol that govern the communication among the several agents, support some processes like: Customer Orders Gathering, Sharing of Customer Orders to dispatch, Sharing of the product related to specific Customer Orders to deliver, Product Delivery to Customer.

Each operation related to these processes can create an anomaly or a faultiness that is registered by a *Ticket.* Examples of tickets are: product (in)availability, invoicing fault, payment block, product exchange, customer exchange…

Every ticket is associated to a code, to a number and type of customer order, to a date of receives, to a description, as show the **Figure 14**, and register an anomaly of the process.

| DATE_ELAB | COD_ANALYSIS | COMPA | Type | NUM_ORDER | Description |
|---|---|---|---|---|---|
| 02-gen-07 | BCK | 1 | SO | 7000690 | Product availability |
| 02-gen-07 | BCK | 1 | SO | 7000693 | Product availability |
| 02-gen-07 | BCK | 1 | SO | 7000699 | Product availability |
| 02-gen-07 | BCK | 1 | SO | 7000702 | Product availability |
| 02-gen-07 | VETTORI | 1 | SO | 7000704 | Carries Service Level |
| 02-gen-07 | BCK | 1 | SO | 7000705 | Product availability |
| 02-gen-07 | BCK | 1 | SO | 7000706 | Product availability |
| 02-gen-07 | P36C | 1 | SO | 7000707 | Order Fault by Commercial Agent |
| 02-gen-07 | P37C | 1 | SO | 7000707 | Invoicing Fault |
| 02-gen-07 | BCK | 1 | SO | 7000710 | Product availability |
| 02-gen-07 | P36C | 1 | SO | 7000712 | Order Fault by Commercial Agent |
| 02-gen-07 | P37C | 1 | SO | 7000712 | Invoicing Fault |
| 02-gen-07 | BCK | 1 | SO | 7000715 | Product availability |
| 02-gen-07 | VETTORI | 1 | SO | 7000715 | Carries Service Level |
| 02-gen-07 | BCK | 1 | SO | 7000716 | Product availability |
| 02-gen-07 | BCK | 1 | SO | 7000718 | Product availability |
| 02-gen-07 | BCK | 1 | SO | 7000719 | Product availability |
| 02-gen-07 | P36C | 1 | SO | 7000724 | Order Fault by Commercial Agent |
| 02-gen-07 | P37C | 1 | SO | 7000724 | Invoicing Fault |
| 02-gen-07 | BCK | 1 | SO | 7000725 | Product availability |
| 02-gen-07 | BCK | 1 | SO | 7000726 | Product availability |
| 02-gen-07 | VETTORI | 1 | SO | 7000726 | Carries Service Level |

**Figure 14: Tickets data base**

### *Atomic Indicators*

With the information of tickets it is possible to create a set of performance indicators that measure the percentages of faults with respect to the total of customer orders. The **Figure 15** shows the fault's number with respect to every tickets (k= 1…2), to every phase (j =1…7), to every agent (i=1…3).

| TICKETS | FAULT'S NUMBER | AGENT | PHASE | Ship. Weight | Ware. Weight | Ord. Weight | Qual. Weight | Adm. Weight | Line Weight | Prod.Av. Weight |
|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 15 | Warehouse | Warehouse Management | | 0,1 | | | | | |
| 010 | 2 | Carrier | Shipment Management | 0,02 | | | | | | |
| 02 | 7 | Warehouse | Warehouse Management | | 0,05 | | | | | |
| 03 | 31 | Warehouse | Warehouse Management | | 0,3 | | | | | |
| 033 | 3 | Warehouse | Warehouse Management | | 0,01 | | | | | |
| 05 | 8 | Warehouse | Warehouse Management | | 0,03 | | | | | |
| 06 | 34 | Company | Product Quality | | | | 0,3 | | | |
| 07 | 5 | Warehouse | Warehouse Management | | 0,08 | | | | | |
| 077 | 16 | Warehouse | Warehouse Management | | 0,3 | | | | | |
| 08 | 1 | Warehouse | Warehouse Management | | 0,05 | | | | | |
| 080 | 5 | Warehouse | Warehouse Management | | 0,08 | | | | | |
| 089 | 2 | Carrier | Shipment Management | 0,03 | | | | | | |
| 09 | 7 | Carrier | Shipment Management | 0,04 | | | | | | |
| 090 | 1 | Carrier | Shipment Management | 0,03 | | | | | | |
| 091 | 3 | Carrier | Shipment Management | 0,05 | | | | | | |
| 10 | 2 | Carrier | Shipment Management | 0,05 | | | | | | |
| 11 | 21 | Carrier | Shipment Management | 0,2 | | | | | | |
| 12 | 179 | Company | Product Quality | | | | 0,7 | | | |
| 18 | 828 | Company | Orders Gatting | | | 0,9 | | | | |
| 19 | 824 | Company | Administration | | | | | 0,5 | | |
| 21 | 9 | Company | Orders Gatting | | | 0,1 | | | | |
| 39 | 3 | Carrier | Shipment Management | 0,04 | | | | | | |
| 41 | 1 | Carrier | Shipment Management | 0,04 | | | | | | |
| BLOCK | 747 | Company | Administration | | | | | 0,5 | | |
| CT UNAVAIL | **13707** | Company | Product Availability | | | | | | | 1 |
| RVICE LEV | 1834 | Carrier | Shipment Management | 0,5 | | | | | | |
| LINE OUT | 290 | Warehouse | Righe Tagliate | | | | | | 1 | |

**Figure 15: Fault numbers**

The concept of phase can be explicated with the use of UML diagram formalism. The **Figure 16** displays the four phases that concern the company operations.

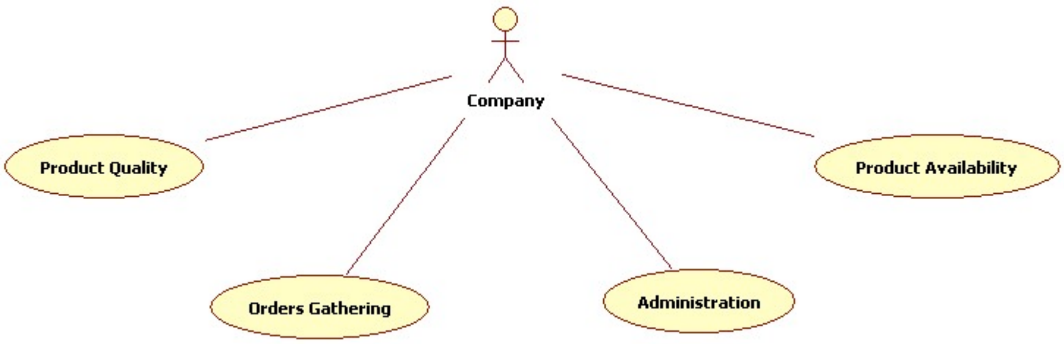


**Figure 16: Use case diagram for Company phases**

Starting from this amount of data it is possible build a set of indicators according to customer satisfaction, then respect to total of customer orders that the company received in a specific time slot. The set of atomic indicators, with corresponding ticket and formula is given in the **Figure 17**.

| Agent | Phase | Description | Formula | Indicators | Performance |
|---|---|---|---|---|---|
| Wharehouses | Wharehouse Management | Numbers of orders that contains lines with these kind of tickets: 01,02,03,033,05,07,077,08,080 | Numbers of orders with a wharehouse's fault/Total number of orders | 0,00002845 | 0,99997155 |
| Carriers | Shipment Management | Numbers of orders that contains lines with this kind of tickets: 010,089,09,090,091,10,11,39,41,Service Level | Numbers of orders with a shipment's fault / Total number of orders | 0,0015367 | 0,9984633 |
| Company | Product Quality | Numbers of orders that contains lines with this kind of tickets: 12,06 | Numbers of orders with a quality fault / Total number of orders | 0,000225833 | 0,999774167 |
| Company | Orders Getting | Numbers of orders that contains lines with this kind of tickets: 18,21 | Numbers of orders with an order getting's fault / Total number of orders | 0,0012435 | 0,9987565 |
| Company | Administration | Numbers of orders that contains lines with this kind of tickets: 19,Block | Numbers of orders with an amministration's fault / Total number of orders | 0,001309167 | 0,998690833 |
| Company | Product availability | Numbers of orders that contains lines with this kind of tickets: product unavailability | Numbers of orders with an availability's fault / Total number of orders | 0,022845 | 0,977155 |
| Wharehouses | Line out | Numbers of orders that contains lines with this kind of tickets: line out | Numbers of orders with a line out's fault / Total number of orders | 0,000483333 | 0,999516667 |

**Figure 17: Set of atomic indicator**

*Agents and Supply Chain Indicators*

Appling the formula *(5)* is possible to obtain an indicator that measure the performance of each agent with respect to fault's data base built. In particular the value indicators are the result of following operation:

1. Company Performance = Product of value indicators related to phases: Product Quality, Orders Getting, Administration, Product Availability = 97,444%.

2. Carriers Performance = Value indicator related to Phase Shipment Management = 99,949%.
3. Warehouses Performance = Product of value indicators related to phases: Warehouses Management and Line Out = 99,846%.

The integrated SC indicator is output of product among agent's performance, then:

4. Supply Chain Performance = Product of value indicators related to agents Company, Carriers and Warehouses = 97,24%.

Finally, in order to offer:

- an important possibility to improve the organization and the performance of the whole Supply Chain;
- a measure to Company's credibility to present to the new customers,

an easy dashboard that report the company performance trends is presented.

### 4.3.5 Conclusion

In this work I propose a Model, called PIM, able to support the Integrated Supply Chain Manager in the Indicator's System building with the use of the Multi-Agent System theory. In particular, I present a real case study that demonstrate the validity of the model with respect to ability of evaluation of Supply Chain's agents performance. In the next works we would test the model on a complete chain, introducing an exhaustive analysis about the supplier's role. At the same time, we are going to develop an information technology platform in order to offer an indicator system on-line for the Company object of case study here presented.

## 4.4 Territorial Collaborative Networks

### 4.4.1 Introduction

The rapid evolution in customer requirements is forcing major changes in the overall industrial system. One possible strategy for facing these changes is based on the adoption of collaborative way of working where different abilities and competences are brought together with the goal of exploiting benefits and sharing the risks. This idea is supported by the increasing relevance of the collaboration as new multi disciplinary research field (Camarinha-Matos and Afsarmanesh, 2004).

This work aims at providing a model for understanding the dynamics of a network composed of heterogeneous actors, and suggests a competence-based collaborative way of working, where the competence is defined as the ability to perform activities by using a combination of knowledge, skill and attitude. As Camarinha-Matos and Afsarmanesh (2006) argue in their work, the definition of a model certainly represents one of the main topics concerning the collaborative network organization research field.

The model well represents actors working on a given geographical area, where the area boundaries are both physical, and due to the existence of consolidated business connections among the actors (Albino et All, 2005). In particular, the model represents a breakthrough with respect to the work by Confessore, Liotta, and Rismondo (2006), where the authors exploited the concepts of "competence measure" and "competence map" to solve the problem of assigning to collaborative enterprises the activities required for carrying out an emerging business process. They provided a Multi Agent System -MAS- (see Jennings and Wooldridge, 1995), in which the actors share information about their degree of competence in doing the activities without reveal private data. In fact, the competence there represent an aggregate data based on a local evaluation, and all the actors measure themselves with respect to the same set of competences given by the competence map (see also Hammami, Burlat, and Champagne, 2003). On the basis of the previous MAS, the new model considers new agent typologies and new features for the decision-making processes in order to allow the evaluation of the impact of new configurations of the network with respect to key performance indicators. The new configurations of the network are generated whenever emerging business processes and possible public funding (e.g., as calls for National and European research project) arise, and it is required to define the roles and responsibilities through the actors.

### 4.4.2 The Scenario

The application provides a model for the understanding of the dynamics of a collaborative network. The network is represented by a coordinator, and private actors. The coordinator makes decisions for increasing the territorial attractive capacity respect to new investments and new projects. Its main tasks are:

1. The monitoring of new business opportunities by doing intensive market analysis. The output are: the proposal of activities to the agents in order to meet the business opportunities; the identification of new possible attractive industrial sectors that could be exploited with the actual territorial resources;
2. The monitoring of call for National and International research projects. The output is to suggest possible combinations of actors in order to create the suitable composition of partners meeting the call requirements;
3. Providing to the actors the competence map in order to meet both the business and funding opportunities while using a competence-based criterion as a way for comparing the actors' capability.

Due to the competence-based criterion, each private actor has the main goal of increasing its competences. Indeed, this condition allows it to obtain an increasing number of activities of an emerging business process or it allows to become an eligible actor for a research project as a partner. A greater number of activities and of research project participations, generate greater revenue for the private actor that could be invested again for increasing the competences by generating a positive feedback.

Summarizing, the competence-based criterion driving either the business process activities assignment, and the public funding exploitation, generates a process of continuous development of the territorial competences, stimulating the collaboration between private actors. Moreover, the investments growth pulls new research projects and business processes by attracting also new enterprises actors that are motivated to join the collaboration or to work in the profitable territory.

For measuring the benefits given by the competence-based collaborative network, the following Social Wellness indicators are suggested:

- Number of new research projects/business processes approved and finished. These indicators represent a measure of the territory attraction with respect to research project and business processes, respectively.
- Number of new competences characterizing the territory. The set of competences is not static since new competences can be required, and others can be not more useful to realize a research project or a business process. This indicator is useful for understanding how the competences of the network evolves.
- Number of actors operating on the territory. It indicates the development of the collaborative network with respect to the actors' composition.

### 4.4.3 The Multi Agent System

The MAS is composed of classes of interacting agents each one having its local information and goals. In this setting, the decision-making processes and the competences characterize in a specific way each one of the different agents working in the system. In particular, it is supposed the agents being represented by two distinct typologies: (*i*) the Territorial Agent (TA), and (*ii*) the Enterprise Agent (EA). TA represents the coordinator, while each EA (e.g., representing a private company) interacts with the other agents in order to pursue its goals. The following notations are used through the paper:

- Let $E = \{e_1, e_2, \ldots, e_z\}$ be the set of $z$ EAs.
- Let $A = \{a_1, a_2, \ldots, a_k\}$ be the set of $k$ activities in which a business process or a research project can be decomposed.
- Let $C = \{c_1, c_2, \ldots, c_w\}$ be the competence map, that is a set of $w$ competences globally accepted by all the agents.

Next, the decisional processes are explained.

*Territorial Agent*

According to the described scenario, TA acts as a coordinator and it solve the decisional problem called Assignment Problem (AP) in order to assign each activities to exactly one of the EA.

**The Assignment Problem**

By solving the Assignment Problem, TA obtains the efficient allocation of the activities of a business process to the EAs. The parameters $\omega_{ij}$ represents the capability of $e_j \in E$, with respect to the execution of the activity $a_i \in A$. These values are provided by each EA and are computed as described in the next Section 3.2.1. Given the binary decision variable $x_{ij}$ equal to 1 if the activity $a_i$ is assigned to the enterprise agent $e_j$, and 0 otherwise, it is possible formulizing the AP as following:

$$\begin{aligned} \text{Min } & z = \sum_{i\in A}\sum_{j\in E} \omega_{ij}\, x_{ij} \\ \text{s.t } & \sum_{j\in M\cap E} x_{ij} = 1 \quad \text{for all } a_i \in A \end{aligned} \tag{8}$$

*Enterprise Agent*

Each EA has to: (*i*) be able to define its degree of competence respect to the competences required by the activities of a business process; (*ii*) solve the decisional problem of choosing which typology of investment select in order to increase its competence, that can be referable to a special case of the Capital Budgeting Problem (Tobin, 1999); (*iii*) decide if a coordinator proposal can be profitable or not.

**Competence evaluation**

Each EA has to evaluate its degree of competence with respect to each activity, thus it has to define the value $\omega_{ij}$. This value can be computed as described in Confessore, Liotta, Rismondo (2006) by modelling the subsets of competences required by an activity and declared by an actor as vectors. The value of $\omega_{ij}$ is obtained by computing the Euclidean distance between the given vectors.

**Capital Budgeting Problem**

The problem is to select the investments maximizing the total return in term of competence while respecting the budget constraints. The best solution is the one producing the maximum competence increment. The following parameters are introduced:

- Let $L$ be the set of possible investments.
- Let $Q(e_j)_{[h,i]}$ be a matrix of parameters where the generic element $q_{hi}(e_j)$ represents the return of competence $c_h \in C$ for $e_j \in E$, respect to the investment $i \in L$.
- Let $\varepsilon_{jh}$ be a positive value in [0, 1] representing the degree of competence of the enterprise $e_j$ respect to a specific $c_h \in C$.
- A budget $B(e_j)$ representing a monetary value, for all $e_j \in E$.
- A cost of investment $b_i(e_j)$, for all $i \in L$, and $e_j \in E$.

Given the binary decision variable $x_i$ equal to1 if the investment $i$ is selected, and 0 otherwise, it is possible formulizing the problem above mentioned as following:

$$\text{Max } \; t = \sum_{h\in C}\sum_{i\in L} q_{hi}\, x_i$$

$$\text{s.t.} \quad \sum_{i\in L} b_i x_i \leq B(e_j) \qquad \text{for all } e_j \in E$$

$$\sum_{i\in L} (\varepsilon_{jh} + q_{hi}(e_j)) \leq 1 \quad \text{for all } e_j \in E \text{ and } c_h \in C \qquad (9)$$

The second constraints model the idea that for each competence exists a threshold value equal to 1. This value was introduced in Confessore, Liotta, Rismondo (2006) as the maximum value for doing as best one activity given the common scale of benchmark values.

Once the decisional problem is solved, the EAs update their competences, that is $\varepsilon_{jh}' = \varepsilon_{jh} + q_{hi}(e_j)$ if $x_i$ equal to1 thus the investment $i \in L$ is selected, for all $e_j \in E$ and

$c_h \in C$. Noteworthy, the system dynamics is based on the hypothesis that if the actors do not invest in a competence for long time, the value $\varepsilon_{jh}$ decreases.

**Project Partecipation Evaluation**

Each EA decides to participate to a business process or to a research project based on the following data:

- Capacity availability.
- Profitability with respect to the increment of the degree of competences.
- Profitability with respect to the collaboration. For instance the collaboration with other actors could remain also when the project ends.
- Project relevance at scientific and research levels.

### 4.4.4 The Multi Agent System Dynamic

This Section summarizes the main features of the interaction protocol exploited by the agents, then defining the dynamics of the MAS. It is supposed that the agents react in response to two possible events, that is an emerging business or a funding opportunity arise. The result of each decisional problem contributes to the definition of a new system configuration.

*Business Process*

Whenever a business opportunity occurs, two levels of interaction between the agents can be defined.

*First level*: The information flow is from TA to the EAs and vice versa.

Information Domain: TA manages a list of codified competences (i.e. the competence map, globally accepted by the agents), a list of agents operating in the system. TA decomposes the business process in a set of activities, and defines the subset of competences required for each activity.

Goal: TA has to allocate each activities of the business process according to competence-based criterion, thus maximizing the total degree of competences for realizing the project.

Communication Protocol: TA communicates to the EAs the set of activities and the related set of competences required for doing them. Each EA then communicates to TA an aggregate data describing its level of competences.

*Second level*: It corresponds to the EAs actions in response to the business opportunities. Since at the first level TA decides by using the competence-based criterion, each EA has to improve its degree on competence by selecting possible investment.

Information Domain: Each EA knows the competence map.

Goal: Each EA solves the problem of selecting from a set of profitable investment the sub-set of them maximizing the return of competence while satisfying its budget.

Communication Protocol: Each EA communicates its availability in executing the activities, or its degree of competences in order to stimulate new collaborations

*Funding Opportunities*

Also in this scenario, whenever a funding opportunity occurs, two levels of interaction between the agents can be defined. In this paper, the funding opportunities arise when the TA observes a call for research project.

*First level*: The information flow is from TA to the EAs. It is important to notice that for the EAs the participation to a research project can be view as an alternative profitable investment.

Information Domain: TA manages a list of codified competences, a list of agents acting on the system. Furthermore, TA knows the competences that best suite a call for project, and codifies the composition of agent typologies that have the greater probability of obtaining the financial fund approval.

Goal: TA has to decide the best composition of agents.

Communication Protocol: TA contacts the agents for proposing the project participation.

*Second level*: it corresponds to the EA and PA actions in response to the funding opportunities.

Information Domain: each agent knows its degree of competences.

Goal: EAs aim at carrying out the activities of the research project by collaborating.

Communication Protocol: The EAs response to the TA request by communicating their availability to execute the project activities. If they decide to participate then communicate to TA their competences. The probability of obtaining the public funding will be a function of the agents composition and on the total degree of competences. The agents collaborate during the project duration and they have a return of competences due to the research project collaboration.

### 4.4.5 Conclusions

The application analyses a competence-based collaborative network, identifying roles, decision making processes and the interaction protocol between the actors. The model is based on the Multi Agent System paradigm and it is driven by the competence concept. Even if the model does not capture all the aspects of the collaboration, it represents a further step toward the representation of the collaborative networks, and the understanding of what and how a network of actors has benefits from the collaboration. Actually, both the dynamics and the decisional problems are faced by the implementation of ad hoc algorithms. In the future, the plan is to add new features to the MAS in order to suggest the model as a valid way for studying the complex connections between collaborative actors, and then to exploit it in a real case-study as preliminary done in Baffo et al. (2006).

# 5

# CONCLUSION

Managing and control distributed system is complex like so designing and building agent systems is difficult. They have all the problems associated with building traditional distributed, concurrent systems and have the additional difficulties that arise from having flexible and sophisticated interactions between autonomous problem-solving components. The big question then becomes one of how effective MASs can be designed and implemented.

At this time, there are two major technical impediments to the widespread adoption of multi-agent technology:

(i) the lack of a systematic methodology enabling designers to clearly specify and structure their applications as MASs and (ii) the lack of widely available industrial-strength MAS toolkits. Flexible sets of tools are needed that enable designers to specify an agent's problem-solving behaviour, specify how and when agents should interact, and visualize and debug the problem-solving behaviour of the agents and the entire system.

The other major impediment to the widespread adoption of agent technology has a social, as well as a technical, aspect. For individuals to be comfortable with the idea of delegating tasks to agents, they must first trust them. The process of mutual adjustment between user and agents (both in terms of the agent learning user preferences but also in terms of the user learning agents' capabilities and limitations) takes time. During this period, agents must strike a balance between continually seeking guidance (and needlessly distracting the user) and never seeking guidance (and exceeding their authority).

For these reasons the outcome of my researches can be presented as a reference model integrating two types of approach for the models creation in the MAS field. The main scope is to provide the guidelines able to support the designer in the system modelling like Multi Agent System, indicating also in which scenario is more convenient to adopt an approach oriented to Operation Research (OR) technique or another one. The Reference Model is evaluated and validated thanks to several application in different contests, such as, flexible manufacturing systems, health care systems, territorial productive systems and integrated logistic systems.

The results have shown that the Reference Model's use can support the designer or the decision maker to built model able to manage and represent complex system in several contests. Every system is represented as a set of intelligent entities able to solve local problem and to communicate with other entities in order to reach a more global goal. In my research, I tried to combine these Multi Agent System theory's principles with the possibility offered by Operation Research techniques with respect to problem solving field.

# 6

# REFERENCES

Adacher L, Agnetis A, Meloni C, 2000, Autonomous agents architectures and algorithms in flexible manufacturing systems, IIE Transactions, Vol.32, pp. 941-951.

Adacher L, Agnetis A, and Meloni C, 2000, Autonomous agents architectures and algorithms in flexible manufacturing systems, IIE Transactions Vol. 32, pp. 941-951.

Akcali E, Cote´ MJ, Lin CA, 2006, Network flow approach to optimizing hospital bed capacity decisions, HealthCare Manage Science, Vol. 9, pp. 391-404.

Albino V, Carbonara N, Giannocchiaro I, 2005, Organizzazioni e Complessità. Muoversi tra ordine e caos per affrontare il cambiamento, Franco Angeli Editore, Milano.

Aptel O, Pourjalali F, 2006, Improving activities and decreasing costs of logistics in hospitals. A comparison of U.S. and French hospitals, The International Journal of Accounting, Vol. 36, pp. 65-90.

Athanassopoulos A, Gounaris C, 2001, Assessing the technical and allocative efficiency of hospital operations in Greece and its resource allocation implications, European Journal of Operational Research, Vol. 133, pp. 416-431.

Baffo I, Calvano A, Confessore G, Valenzi E, 2009, La performance? Su misura del cliente. Come si misura l'eccellenza e come viene percepita, Il Giornale della Logistica, pp. 48-50, Aprile 2009.

Baffo I, Confessore G, Galiano G, Rispondo S, A Multi Agent System Model to study a collaborative network, Proceedings of Regional Studies Association International Conference (Abstract), Lisbon (Portugal), April 2007.

Baffo I, Confessore G, Graziano G, Rismondo S, 2008, A multi agent system model to evaluate the dynamics of a collaborative networks, Proceedings of the 10th International Conference on Enterprise Information Systems (ICEIS 2008), Barcelona (Spain), June 2008. ISBN: 978-989-8111-48-7.

Baffo I, Confessore G, Liotta G, Stecca G, 2008, A Reference Model for Distributed Decision Making Through A Multi-Agent Approach". Proceedings of 9th IFIP Working Conference on Virtual Enterprises (Pro-VE) September, 8-10, 2008, Poznan, Poland. In "IFIP International Federation for Information Processing", Volume 283; Pervasive Collaborative Networks; Luis M. Camarinha-Matos, Willy Picard (Eds); (Boston: Springer), 2008, pp. 285–292, ISBN 978-0-387-84836-5, ISSN 1571-5736 (Print) 1861-2288 (Online).

Baffo I, Confessore G, Rismondo S, 2007, 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.

Baffo I, Confessore G, Stecca G, 2007, Development Of A Multi-Agent Model For Production Scheduling In Innovative Flexible Manufacturing Systems, Proceedings of 40th CIRP International Conference on Manufacturing System, Liverpool (United Kingdom), June 2007.

Baffo I, Confessore G, Stecca G, 2008, A Performance Indicators Model to support the Organization of an Integrated Supply Chain, Proceedings of APMS 2008 - IFIP International Federation for Information Processing, Advances in Production Management System, Espoo (Finland), September 2008, in "Innovation in Networks", Riitta Smeds (Ed), Published by Helsinki University of Technology Simlab, 2008, pp. 45-53, ISBN 978-951-22-9547-0.

Baffo I, Confessore G, Stecca G, 2008, Development of a Multi-Agent Model for production scheduling in innovative flexible manufacturing system, Proceedings of 41th CIRP International Conference on Manufacturing System, Tokyo (Japan). In "Manufacturing Systems and Technologies for the new Frontier", Mamoru Mitsuishi Dr.Eng., Kanji Ueda Dr.Eng., Fumihiko Minura Dr.Eng. (Eds), Springer London, 2008, pp. 279-282, ISBN 978-1-84800-267-8, DOI 10.1007/978-1-84800-267-8.

Baffo I, Confessore G, Turina M, 2009, A performance indicators model to shape the geographical clusters development, Proceedings of 15th APDR Conference "Networks and Regional Development", Praia (Cape Verde), July 2009.

Baffo I, Confessore G, Turina M, 2009, An indicators framework to evaluate industrial clusters development , 49th Annual Congress of the European Regional Science Association "Territorial cohesion of Europe and integrative planning", Lodz (Poland), August 2009.

Baffo I, Dedonno L, Confessore G, Rismondo S, 2006, Descrizione ex-post di una rete relazionale territoriale e realizzazione di uno strumento per la simulazione dinamica ex-ante, in Proceedings of the XXVII Italian Conference of Regional Science - Impresa, Mercato, Lealtà Territoriale.

Baffo I, Liotta G, Stecca G, Confessore G, 2009, A Cooperative model to improve hospital equipments and drugs management, L.M. Camarinha-Matos et al (Eds): PRO-VE 2009, IFIP AICT 307, pp.43-50, ISBN: 1868-4238.

Baffo I., Confessore G., Turina M., Studio dei processi di cooperazione per lo sviluppo locale, agli atti della XXIX Aisre Conference 2008, Bari (Italy), September 2008.

Baffo I., Confessore G., Turina M., Uno strumento di supporto per l'assegnamento degli aiuti in regime "de minimis" a favore dello sviluppo locale, agli atti della XXX Aisre Conference 2009, Firenze (Italy), September 2009.

Bauer B, Odell J, 2005, UML 2.0 and agents: how to build agent based systems with the new UML standard, Engineering Application of Artificial Intelligence, Vol. 18, pp. 141-157.

Bianco L, Confessore G, Stecca G, 2005, 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, CD-ROM.

Brooks, R. A. 1991. Intelligence without Representation. Artificial Intelligence, Vol. 47(1–3), pp.139–159.

Camarinha-Matos LM, Afsarmanesh H, 2005, Collaborative networks: A new scientific discipline, Journal on Intelligent Manufacturing, Vol. 16, pp. 439-452.

Camarinha-Matos LM, Afsarmanesh H, 2006, A modeling framework for collaborative networked organizations, in IFIP International Federation for Information Processing, Network-Centric Collaboration and Supporting Fireworks, eds. Camarinha –Matos, L., Afsarmanesh, H., Ollus, M., (Boston: Springer), Vol. 224, pp. 3-14.

Camarinha-Matos LM, Afsarmanesh H, 2007, A comprehensive modeling framework for collaborative networked organizations, Journal of Intelligent Manufacturing, Vol. 18, pp. 529-542.

Camarinha-Matos, LM., Afsarmanesh H, 2004, Towards next business models, in Collaborative Networked Organizations: a research agenda for emerging business models, Collaborative Networks: Reference Modeling.

Cilliers P, 1998, Complexity and postmodernism: understanding complex system, Routledge, London.

Confessore G, Liotta G, Rismondo S, 2006, in IFIP International Federation for Information Processing, Volume 224, *Nerwork-Centric Collaboration and Supporting Frameworks*, eds. Camarinha-Matos, L., Afsarmanesh, H., Ollus, M., (Boston: Springer), pp. 121-128.

Corkill D D, Lesser V R, 1983, The Use of Metalevel Control for Coordination in a Distributed Problem- Solving Network. In Proceedings of the Eighth International Joint Conference on Artificial Intelligence (IJCAI-83), pp. 767–770. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Dang J, Hedayati A, Hampel K, Toklu C, 2008, An ontological knowledge framework for adaptive medical workflow. Journal of Biomedical Informatics, Vol. 41, pp. 829-836.

Davis R, and Smith RG, 1983, Negotiation as a metaphor for distributed problem solving, Artificial Intelligence, Vol. 20, pp. 63–100.

De Angelis V, Felici G, Impelluso P, 2003 Integrating simulation and optimisation in health care centre management. European Journal of Operational Research, Vol.150, pp. 101-114.

De Treville S, Smith I, Rolli A, Arnold V, 200, Applying operations management logic and tools to save lives: a case study of the world health organization's global drug facility. Journal of Operations Management, Vol. 24, pp. 397-406.

Durfee E H, Lesser V, 1989. Negotiating Task Decomposition and Allocation Using Partial Global Planning. In Distributed Artificial Intelligence, Vol. 2, eds. L. Gasser and M. Huhns, pp. 229–244. San Francisco, Calif.: Morgan Kaufmann.

El Maraghy HA, 2006, Flexible and reconfigurable manufacturing systems paradigms, International Journal of Flexible Manufacturing Systems, Vol. 17, pp. 261–276.

Field F, Vole V, 1986, A comparative analysis of food patches. Technical Report 86-4, The Brae Institute for Food Research, University of Insh, Insh, UK.

Fox MS, 1992, Integrated Supply Chain Management, Technical Report, Enterprise Integration Laboratory, University of Toronto.

Garrido L, Sycara K, 1996, Multiagent Meeting Scheduling: Preliminary Experimental Results. In Proceedings of the Second International Conference on Multiagent Systems, pp. 95–102. Menlo Park, Calif.: AAAI Press.

Gasser L, 1991, Social Conceptions of Knowledge and Action. Artificial Intelligence, Vol. 47(1–3), pp.107–138

Genesereth M R., Ketchpel, S P, 1994, Software Agents. Communications of the ACM, Vol.37(7), pp. 48–53.

Gomez-Sanz JJ, Pavon, J, Garijo F, 2002, Metamodels for building multi-agent systems, in Proceedings of the 2002 ACM symposium on Applied computing. ACM Press, pp. 37–41.

Hammami A, P. Burlat, and J.P. Champagne. 2003, Evaluating orders allocation within networks of firms, In International Journal of Production Economics, Vol. 86, pp. 233-249.

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.), pp. 191-200.

Harper Paul R, 2002, A Framework for Operational Modelling of Hospital Resources. Health Care Management Science, Vol. 5, pp. 165-173.

Hewitt C, 1986, Offices Are Open Systems. ACM Transactions of Office Automation Systems 4(3), pp. 271–287.

Jennings N, Sycara K, and Wooldridge M, 1998, A Roadmap for Agent Research and Development. Autonomous Agents and Multiagent Systems 1(1). Forthcoming.

Jennings NR, Wooldridge M, 1995, Applying agent technology, Applied Artificial Intelligence, Vol. 9, pp. 357-369.

Kaplan R S and Norton D P, 1998, Balanced Scorecard: Translating Strategy into Action, Harvard Business School Press.

Keen P GW and M S Scott Morton, 1978, Decision Support Systems: An organisational perspective. AddisonWesley Publishing Company.

Kinny D, Ljungberg M, Rao A, Sonenberg E, Tidhard G, and Werner E, 1992, Planned Team Activity. In Artificial Social Systems, eds. C. Castelfranchi and E. Werner. New York: Springer-Verlag.

Lesser, V. R. 1991. A Retrospective View of FA/C Distributed Problem Solving. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 21(6), pp. 1347–1363.

Lewis, C M, Sycara K, 1993, Reaching Informed Agreement in Multispecialist Cooperation. Group Decision and Negotiation Vol. 2(3), pp. 279–300.

Liu DR, Wu I-C, Hsieh S-T, 2001, Integrating SET and EDI for secure healthcare commerce. Computer Standards & Interfaces, Vol. 23, pp. 367-381.

Loose J, Scruffy P, 1982. How to drill holes in water. In Figtree E. and Lemort D., eds., Coping with almost unsolvable problems, South Belgium Press, Bruxelles, B, pp. 315–460.

Marshall A, 1925, Principles of economics. London, UK: Macmillan.

Martinez Lastra JL, Colombo AW, 2006, Engineering framework for agent-based manufacturing control, Engineering Applications of Artificial Intelligence, Vol. 19, pp. 625–640.

Mullen, T., and Wellman, M. P. 1996. Some Issues in the Design of Market-Oriented Agents. In Intelligent Agents II, eds. M. Wooldridge, J. P. Mü ller, and M. Tambe, pp. 283–298. Lecture Notes in Artificial Intelligence 1037. New York: Springer-Verlag.

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

O'Hare G, Jennings N, 1996. Foundations of Distributed Artificial Intelligence. New York: Wiley.

Onestone A, 1990, A computer simulation of human ignorance. Human Stupidity Journal, Vol. 5 (2), pp. 283–315.

Otto A., Kotzab H., 2003, Does supply chain management really pay? Six perspectives to measure the performance of managing a supply chain. European Journal of Operational Research, Vol. 144, pp. 306–320.

Park S, Sugumaran V, 2005, Designing multi-agent systems: a framework and application. Expert Systems with Applications, Vol. 28, pp. 259-271.

Shoham, Y. 1993. Agent-Oriented Programming. Artificial Intelligence*, Vol.* 60(1), pp. 51–92.

Soubie J, Zaraté P, 2005, Distributed Decision Making: A Proposal of Support Through Cooperative Systems, Group Decision and Negotiation, Vol. 14, pp. 147-158.

Sprague RH and ED Carlson, 1982, Building Effective Decision Support Systems. Prentice Hall, Inc., Englewood Cliffs.

Supply Chain Council, 1996, Supply Chain Operation Reference (SCOR) Model, Overview of SCOR version.

Sycara K, Decker K, Pannu A, Williamson M,  Zeng D, 1996, Distributed Intelligent Agents. IEEE Expert , Vol.11(6): pp. 36–46. Tambe, M. 1997.

Thornett, Andrew Martyn, 2001, Computer decision support systems in general practice, International Journal of Information Management, Vol. 21, pp. 39-47.

Tobin RL, 1999, A fast interactive solution method for large capital expenditure selection problems, European Journal of Operational Research, Vol. 116, pp. 1–15.

Trabelsi A, Ezzedine H, Kolski C, 2004, Architecture modelling and evaluation of agent.based interactive system, IEEE Internationl Conference on Systems, Man and Cybernetics.

Van Merode GG, Groothuis S, Hasman A, 2004, Enterprise resource planning for hospitals. International Journal of Medical Informatics, Vol. 73, pp. 493-501.

Wang J, Deng Y, 1999, Incremental modelling and verification of flexible manufacturing systems, Journal of Intelligent Manufacturing, Vol. 10, pp. 485-502.

Zambonelli F, Jennings NR, Wooldridge M, 2003, Developing Multiagent Systems: The Gaia Methodology, ACM Transaction on Software Engineering and Methodology. Vol. 12, No. 3, pp. 317-370.

# APPENDICE 1

# LIST OF FIGURES

# APPENDICE 2

# LIST OF TABLES