



**HAL**  
open science

## **A multiagent architecture for an underground river network of intelligent sensors**

Jean-Paul Jamont, Michel Occello, André Lagreze

► **To cite this version:**

Jean-Paul Jamont, Michel Occello, André Lagreze. A multiagent architecture for an underground river network of intelligent sensors. IEEE Virtual Instrumentation and Measurement Systems 2002, 2002, Anchorage, United States. pp.20-25. <hal-00201571>

**HAL Id: hal-00201571**

**<https://hal.science/hal-00201571v1>**

Submitted on 1 Feb 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# A MULTIAGENT ARCHITECTURE FOR THE INSTRUMENTATION OF AN UNDERGROUND HYDROGRAPHIC SYSTEM

Jean-Paul JAMONT

LCIS/INPG

50 rue Barthelemy de Laffemas Valence Cedex, 26902, FRANCE

Email: jean-paul.jamont@esisar.inpg.fr

Michel OCCELLO

UPMF - Leibniz/IMAG/CNRS

46 Avenue Felix Viallet Grenoble Cedex, 38031, FRANCE

Email: michel.occello@imag.fr

and

André LAGREZE

UPMF - LCIS/INPG

50 rue Barthelemy de Laffemas Valence Cedex, 26902, FRANCE

Email: andre.lagreze@esisar.inpg.fr

***Abstract** – This paper deals with an application of multiagent systems to sensor network management. This wireless communication network will be applied to monitor an underground hydrographic network. We first present the ENVSYS project: its origin and its issue. We then recall the concepts of agent and multiagent systems. We sketch a multiagent system's architecture according to the AEIO method. This multiagent system consists of hybrid agents. We also introduce the ASTRO agent model.*

***Keywords** – multiagent system, hybrid agent, wireless network, sensor networks*

## I. INTRODUCTION

Multiagent systems are well suited for analyzing and designing complex systems such as networks of distributed autonomous entities behaving in an open environment.

The context of the instrumentation of an underground river system involves an open network of intelligent sensors whose cooperation must be monitored in order to insure the best organization.

In this paper we intend to evaluate the contribution of a multiagent approach on such an application.

We shall begin by presenting the origin and the issue of the ENVSYS project, i.e., an underground wireless communication network. We will then introduce multiagent systems in the context of sensor networks. Finally, we will present the state of our experimentations.

## II. THE ENVSYS PROJECT

The purpose of the ENVironment SYStem project is to monitor an underground river sensor network. Let us present the origin of this project and the problems occurring in such an application.

### A. Origin of the project

The ENVSYS project finds its origin in a statement: the measure of the various parameters in an underground river system is a complex task. In fact, the access to this type of underground galleries is difficult : it requires help from speleologists. Besides, the installation of wire communications networks is difficult, especially because the structure of hydrographic system is very often chaotic. Finally, in the case of a radio communication network, the underground aspect complicates wave propagation and for the moment the techniques which are used are not totally mastered.

For some years, systems of radio communication have been introduced. They are generally used by the speleological rescue specialists. These systems are analogic, work with low frequencies and are used mostly for voice transmission.

The general idea of the project is to study the feasibility of a sensor network from the existing physical layer. This will allow wireless instrumentation of a subterranean river system. Such a network would present an important interest in many domains: the study of underground flows, the monitoring of

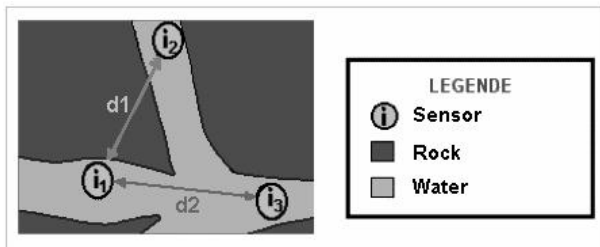
deep collecting, flooding risk management, river system detection of pollution risks, etc.

### B. The issue

In a subterranean river system, the interesting parameters to measure are numerous: temperature of air and water, air pressure and if possible water pressure for the flooded galleries, pollution rate by classical pollutants, water flow, draft speed, etc. All this information will be collected at the immediate hydrographic network exit by a work station like a PC. These data will be processed to activate alarms, study the progress of a certain pollution according to miscellaneous measuring parameters, determine a predictive model of the whole network by relating the subterranean parameters measures of our system with the overground parameters measures more classically on the catchment basin.

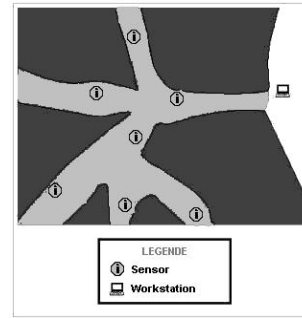
We do not wish to carry out this instrumentation with a wire network for obvious reasons of convenience. We shall use electromagnetic waves with low frequencies as a carrier. These waves have an interesting property: they are able to go through rock blocks. The sensors should then to achieve several functions:

- a measuring function: it is the first function of a sensor. It consists in interacting with the environment to acquire information.
- a relay function: every sensor has a limited transmission range. This limitation results from three points: the technological solutions which are used to achieve the sensor transmission module (frequency, power, antenna), the implementation of these solutions and, finally, the environment. Indeed, according to the obstacles it will have to go through, the electromagnetic waves will not be usable at the same distance for each direction. The transmission zone will not be modeled by a sphere.



The distance  $d_1$  which separates the sensors  $i_1$  and  $i_2$  is shorter than the distance  $d_2$  which separates the sensors  $i_1$  and  $i_3$ . However, the rock separating the couple of sensors ( $i_1, i_2$ ) will degenerate the signal and will prevent sensor  $i_2$  from receiving the message correctly unlike  $i_3$ .

Having defined the role of sensors, we can represent the structure of our communication network. It consists of a set of sensors and a listening station as illustrated on the following figure:



Here is a non-exhaustive list of problems which one needs to address:

- How to realize the physical layer?
- What level of protocol connection to choose above such physical layer?
- How to route the information in the best way? Each of the sensors cannot physically communicate with the workstation which collects the information. Which sensor should thus make the decision to repeat the information?
- How to monitor such a complex environment?
- What kind of intelligence to give to the network?

In the following we will deal with the analysis of the problem using a multiagent system approach. The main contribution of the work presented in this paper is situated at a logical level, concerning especially the last three points of the problems listed before.

## III. MULTIAGENT SYSTEMS AND SENSORS NETWORKS

### A. Multiagent systems

Multiagent systems are a collection of several agents. It is necessary to start by defining what we call an agent. Once this notion has been introduced, we shall approach the concept of MAS.

For the past few years the resolution agents have undergone a strong and fast development of researches. Agent is a generic name which relates to various entities [1]. In fact, it can be biological entities (the associated agents are called biological agents), autonomous robots (robotic agents) or computer software and their components which can be integrated into operational systems or complex computer systems. In the world of distributed artificial intelligence, there is currently no common definition. The confrontation of several definitions [2][3][4] allows us to say that, in our context, " an agent is a software entity embedded in an environment which it can perceive and in which it acts. It is endowed with autonomous behaviors and has objectives ". Autonomy is the main concept in the agent issue: it is the agent's ability to have control over their actions and their internal states. The autonomy of agents implies no centralized control.

The power of an agent decomposition is the decentralization of the intelligence, i.e. the decision capabilities, and of entities' knowledge.

A multiagent system is a set of agents situated in a common environment, which interact and attempt to reach a set of goals. Through these interactions a global behavior, more intelligent than the sum of local multiagent system component intelligence can emerge. The emergence process is a way to obtain dynamic results that cannot be predicted beforehand.

A multiagent system can have several qualifiers. It is open if the system tolerates for the agents to enter and leave freely the multiagent system as opposed to close when the number of agents is always the same. We call homogenous multiagent, in opposition to heterogeneous multiagent system, a system constituted of homogenous agents from the point of view of their theory (representation and properties) and their architecture (particular methodology of agent construction).

### B. AEIO method

The multiagent methods aim at decreasing the complexity of system design by decentralized analysis.

There are several multiagent system methods [5] among which most are centered on the analysis of the agents' tasks as the methods Gaia [6] and MaSE [7], either on the roles or on the organization as the method AALADIN [8]. We are thereafter going to be interested in the AEIO decomposition [9]. We will follow the complete method of analysis and multiagent design discussed in [10], associated to this MAS decomposition. It proposes a decomposition according to four axes collectively accepted today:

- The agent axis (A) gathers all elements for defining and constructing these entities.
- The environment axis (E). This part of the analysis deals with elements necessary for the multiagent system realization such as the perception of this environment and the actions one can do on it.
- The interaction axis (I) includes all elements which are in use for structuring the external interactions among the agents (agent communication language, interaction protocols)
- The organization axis (O) allows to order agent groups in organization determined according to their roles.

We chose to apply this multiagent method for our problem because it privileges an explicit description of the interactions and the environment. In our case, this method will be more adapted than the previously mentioned approaches .

### C. Sensor network management and MAS

Many domains of network administration use intelligent components [11] : configuration management, security management, fault management, performance management and accounting management. This intelligence usually comes from

distributed artificial intelligence.

The distributed and open nature of sensor networks makes the multiagent system approach an adapted answer. Another advantage of it is the external representation of their interactions and the organization which offers multiple possibilities such as the monitoring by an external observer.

A few works reaching the same objectives show that the approach is interesting. We can quote the ActComm [12] project which is a military project for which the routing of information is essential: it aims at studying the communication management between a soldier team and a military camp via a satellite. Another example is the Unmanned Ground Vehicle Program ARPA's project [13] which approaches the information management resulting from a group of autonomous observation military vehicles.

We can find some similarities between EnvSys and ActComm, especially in the aspects of maintenance of network connectivity and the use of network links (see APRL (Any Path Routing with Loops) [14] and GPSR (Greedy Perimeter Stateless Routing) [15]). But we must take into account stronger real-time constraints. We also have to consider messages priority. Some ideas from ARPA for the cooperative sensor planning can be interesting in our case.

## IV. MULTIAGENT SYSTEM APPLICATION TO ENVSYS

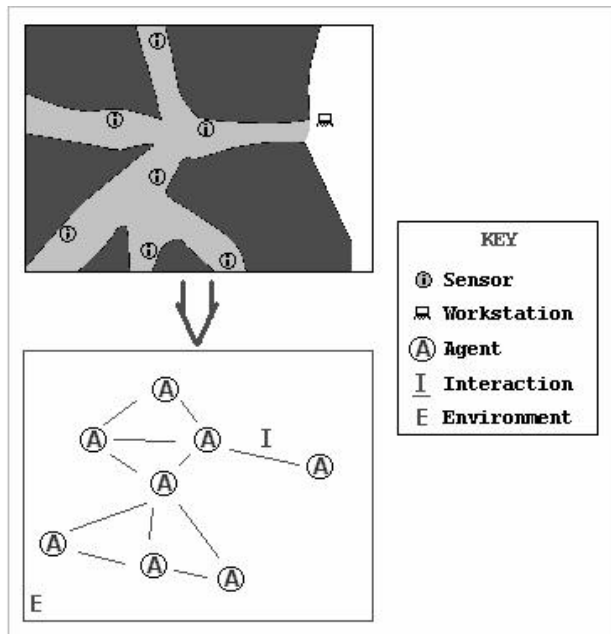
### A. AEIO analysis of ENVSYS

As previously examined, this approach is articulated around four axes.

- The agent axis: In our multiagent system sensors are modeled by agents. These agents have hybrid architectures, i.e. a composition of some types of architecture. Indeed, the agents will be of a cognitive type in case of a configuration alteration, it will be necessary for them to communicate and to manipulate their knowledge in order to have an efficient collaboration. On the other hand, in normal use it will be necessary for them to be reactive (a reactive agent reacts to a stimulus by a response) to be most effective.
- The environment axis: The environment will be made of all the measurable information by every agent and by the spatial coordinates they cannot perceive.
- The interaction axis: The agents will interact only with the agents in acquaintance (an agent is in acquaintance with another one if it is aware of its existence). They interact by message exchange. We can distinguish two methods for these communications: a synchronous and an asynchronous version. In the former, agents must be in rendezvous. On the latter, the asynchronous version, messages are memorized. It is not necessary for the receiver and the sender to be synchronized. Our application having no particular constraint from the communication point of view, we opt for an asynchronous communication by messages

sending which is the most flexible method. Among the different protocols that we use, the choice of an introduction protocol is essential. Indeed, this protocol allows to the agents be known, i.e. to bring their knowledge and their know-how to the agents' society.

- The organization axis: The main property of this organization is that it will be dynamic. In this type of application one cannot control the organization a priori. Relations between agents are going to emerge from the evolution of the agents' states and from their interactions. We are going to be content with fixing the organization parameters, i.e. agents' tasks, agents' roles and initial organization structure.
- Result of this approach:



### B. Agent architectures

The choice of an agent architecture is very important in our problem. We shall start with a reminder on the main agents architecture then we shall introduce ASTRO, the model which we chose.

The sources of reactive agents architecture are the sciences of nature and life. This type of agents works according to the stimulus/response principle. They do not have a clear representation of their environment and do not have memory. One of the most popular examples of the use of this type of agents is that of the ant-hill [16].

Cognitive agent architecture arises from the human model and leans on human and social sciences. This type of agents has a real representation of their environment, the other agents and of themselves. Besides they are endowed with capacity of reasoning and with economic planning motivated by their own purposes.

Using a hybrid architecture for the agents enables to combine

the strong features of each of them. This is especially used for constraints such as real time setting.

ASTRO hybrid architecture [17] [18] is especially adapted to a real time context. The integration of deliberative and reactive capabilities is possible through the use of parallelism in the structure of the agents. Separating Reasoning/Adaptation and Perception/ Communication tasks allows a continuous supervision of the evolution of the environment. The reasoning model of this agent is based on the Perception/Decision/Reasoning/Action paradigm. The cognitive reasoning is thus preserved, and predicted events contribute to the normal progress of the reasoning process.

Decision modules evaluate the importance of unpredicted events and have the obligation to place new actions or new goals in the internal state of the agent's reasoning. New goals imply the activation of the reasoning modules in order to partially or totally replan according to the importance of the event.

This model comprises its knowledge about the environment, the internal states of other agents, and its own internal state. The model is maintained by an interpretation process of the sensory data.

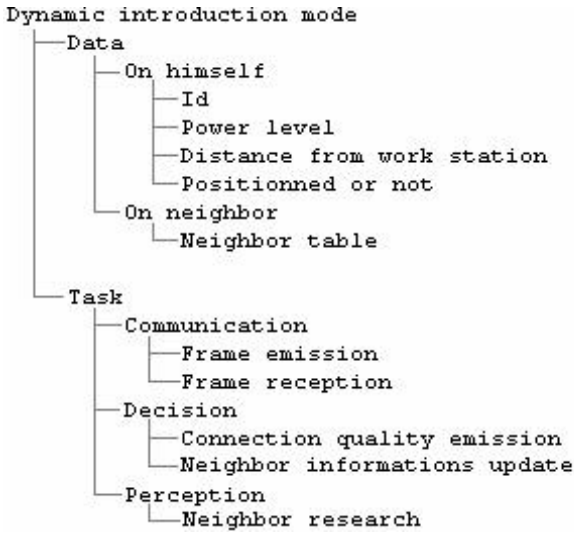
Evolving in a real world, each agent has to integrate perception capabilities achieved through sensor devices. The knowledge of the environment is constructed by the perception modules. Other agents are "perceived" through communication modules. Agents can send information about their knowledge of the environment, their plans, their goals or their current state. Communication modules are probe loop events waiting for messages from agents. Emitters are considered as actions.

To ensure the reactivity of the agent, an evaluator continuously examines the world model. Agent control modules detect situations to which the agent needs to react, evaluate them, and decide to take the appropriate actions which may be to create, suspend, or kill goals, i.e. to change the context of the planning and executing process. The continuous supervision of the agent's situation ensures that the agent can react to unpredicted events.

### C. Task analysis

We can already draw an agent task draft specification. In fact, we can identify two different modes: the first one corresponding to the dynamic introduction of mobile intelligent sensors (agents); the second one consists in a normal measuring phase mode. In terms of agent; the first mode uses an introduction protocol technique, and the second one involves dynamic integrity maintenance mechanisms.

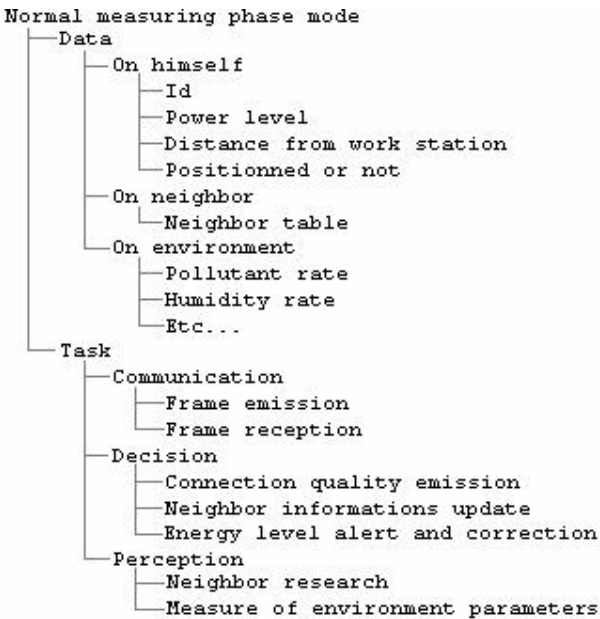
In the configuration mode an agent checks its own state and also its neighborhood state. An agent of this type can accomplish these different tasks:



Note:

- The distance unit between an agent and the work station is the number of relay agent
- The position information is memorized through a boolean variable.
- The neighbor table contains a lot of information about the agent neighbor (id, distance, connection quality).

In normal measuring mode, an agent can accomplish these difference tasks:



## V. TOWARD AN OPERATIONAL EMBEDDED ARCHITECTURE

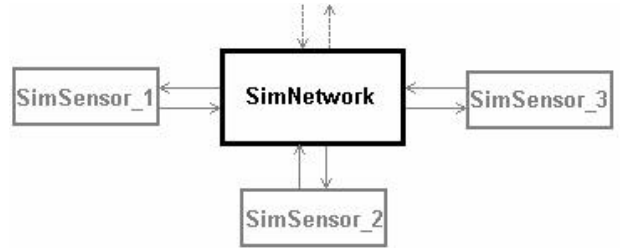
We can consider that we use agents as decisional structures for the intelligent sensors. In order to evaluate and improve such

agents' software architectures and the cooperation techniques that they involve, we introduce a simulation stage in our development process. In this section, we describe this simulation step and then give an insight to the operational embedded architecture.

### A. Simulation

The simulation first allowed us to experiment our approach and the software solutions that we provide for the various problems. We can also quantify the emergence inferred by the MAS approach in this case.

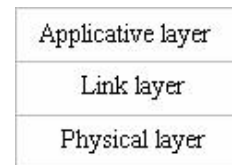
The simulation software structure is very basic. In fact, we have two types of components: SimSensor and SimNetwork.



A SimSensor component simulates the sensor behavior. It possesses its own model and architecture. All the sensors have the same communication capabilities. They transmit their requests to the SimNetwork component sends this information to all sensors which can receive them, in the environment. SimNetwork can appear as the inference mechanism for the simulation.

### B. The operational embedded architecture

We currently implement a minimal system in a river system in Vercors (French Alps) with three sensors to measure the air temperature. A master station situated at the cavity exit will collect this information. Therefore, we will demonstrate the feasibility of our approach. For the sensors we have chosen a classical three-layer architecture of the following type:



We use the physical layer which is employed by NICOLA system, a voice transmission system used by the French speleological rescue teams [19]. This layer is implemented in a digital signal processor rather than a full analogic system. Thereby we can keep good flexibility and we are able to apply further a signal processing algorithm to improve the data transmission.

The link layer used is a CAN (Controller Area Network) protocol stemming from the motorcar industry and chosen for its good reliability. The applicative layer is constituted by the agents' system. The agents are embedded on autonomous processor cards. These cards are equipped with communication modules and with measuring modules to carry out agent tasks relative to the instrumentation. These cards supply a real time kernel. The KR-51 (the kernel's name) allows multi-task software engineering for C515C microcontroller. We can produce one task for one capability. We can then quite easily implement the parallelism inherent to agents and satisfy the real-time constraints.

## VI. CONCLUSION

This software agent architecture is embedded on the autonomous processor cards. The multiagent system which we are creating has two important features. First of all it is an open system: adding a sensor does not require a manual re-configuration. More sensors' dysfunctions should not threaten the functional integrity of the whole system: it should be fault tolerant. Besides, our multiagent system is homogeneous. Indeed, all the sensors will have a hybrid architecture based on the ASTRO model.

Through the simulation step, we can already notice what the MAS approach provides versus a classic approach. We summarize these contributions in three points:

- The emergent feature, which is inferred by the MAS approach, makes the system more flexible and robust. Indeed, the system is more fault tolerant to the changes of the environment in which it evolves. It is more able to process uncertain events than a more classical approach. The resulting system becomes much more adaptive.
- The problem analysis is simplified: one is interested only in basic unit (the agents) and the intelligence of the system will be generated by their interactions.
- Agents present interesting features of software engineering such as genericity allowing an easy evolution of the applications. Furthermore, the generic aspect of the agents allows us to envisage different applications for this network type such as diagnosis, risk management, data fusion...

In a near future, we shall have physical agents which will definitively validate our approach.

## References

[1] S. Franklin and A. Grasser, "Is it an agent, or just a program?: A taxonomy of autonomous agents," in *Proceedings of ATAL 96 - the Third International Workshop on Agent Theories, Architectures and Languages*, 1996, Springer-Verlag.

[2] J. Ferber, *Les Systèmes Multi-Agents : vers une intelligence collective*, InterEditions, 1995.

[3] Y. Demazeau, "Steps towards multi-agent oriented-programming," *1st International Workshop on DAI and MAS*, 1995.

[4] M.J. Wooldridge, "Intelligent agents," in *Multiagent systems: A modern approach to Distributed Artificial Intelligence*, Gerhard Weiss, Ed. 1999, MIT Press.

[5] C. Iglesias, M. Garrijo, and J. Gonzales, "A survey of agent oriented methodologies," in *Proceedings of ATAL 98 - Workshop on Agent Theories, Architectures, and Languages*, Paris, France, July 1998, vol. LNAI 1555, pp. 163–176, Springer-Verlag.

[6] M.J. Wooldridge, N.R. Jennings, and D. Kinny, "The gaia methodology for agent-oriented analysis and design," in *Autonomous Agents and Multi-Agent Systems*, 2000, vol. 3, pp. 285–312, Kluwer Academic Publishers.

[7] M.F. Wood and S.A. DeLoach, "An overview of the multiagent systems engineering methodology," in *The First International Workshop on Agent-Oriented Software Engineering*, 2000.

[8] J. Ferber and O. Gutknecht, "A meta-model for the analysis and design of organizations in multi-agent systems," in *Proceedings of ICMAS'98*, Y. Demazeau, Ed., Paris France, July 1998, IEEE Computer Society.

[9] Y. Demazeau, "From interactions to collective behavior in agent-based systems," in *European Conference on Cognitive Science*, Saint-Malo France, 1995.

[10] M. Occhetto and J.L. Koning, "Multi-agent based software engineering: an approach based on model and software reuse," in *From Agent Theory to Agent Implementation II - EMCSR 2000 Symposium*, Vienna, April 2000, pp. 645–657.

[11] G. Pujolle, *Les Réseaux, 3me dition mise jour*, Eyrolles, 2000.

[12] R.S. Gray, "Soldiers, agents and wireless networks: A report on a military application," in *Proceedings of the Fifth International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agents*, Manchester, England, April 2000.

[13] D.J. Cook, P. Gmytrasiewicz, and L.B. Holder, "Decision-theoretic cooperative sensor planning," *IEEE Transactions on Pattern Analysis And Machine Intelligence*, vol. 18, October 1996.

[14] B. Karp and H.T. Kung, "Dynamic neighbor discovery and loop-free, multi-hop routing for wireless, mobile networks," Tech. Rep., Harvard University, May 1998.

[15] B. Karp and H.T. Kung, "Greedy perimeter stateless routing," in *Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking - MOBICOM 2000*, February 2000.

[16] A. Drogoul, *De la Simulation Multi-agents la Résolution Collective de Problèmes*, Ph.D. thesis, Université Paris VI, November 1993.

[17] M. Occhetto and Y. Demazeau, "Modelling decision making systems using agents for cooperation in a real time constraints," in *3rd IFAC Symposium on Intelligent Autonomous Vehicles*, Madrid, Spain, March 1998, vol. 1, pp. 51–56.

[18] M. Occhetto, Y. Demazeau, and C. Baeijs, "Designing organized agents for cooperation in a real time context," in *Collective Robotics*, A. Drogoul, M. Tambe, and J. Singh, Eds. March 1998, vol. LNCS/LNAI 1456, pp. 25–73, Springer-Verlag.

[19] N. Graham, "The Nicola Mark II - a New Rescue Radio for France," in *The CREG Journal*, December 1999, vol. 38, pp. 3–6.