

Negotiation Between Economically Motivated Agents

E. Oliveira[‡], J. M. Fonseca^{}, A. Steiger-Garção^{*}*

*[‡]FEUP - Faculdade de Engenharia da Universidade do Porto
Rua dos Bragas, Porto, Portugal
Tel +351-2-2004527 E-mail: eco@garfield.fe.up.pt*

^{}UNINOVA - Centre of Intelligent Robotics
Quinta da Torre, 2825 - Monte Caparica, Portugal
Tel/Fax +351-2-3500200/2941253 E-mail: jmf,asg@uninova.pt*

Abstract

The real-time resource management and budget calculation in the civil construction sector is usually a complex problem, specially when involving a large number of heterogeneous entities, geographically distributed over a wide area. Moreover, the environment unpredictability that usually leads to the necessity of a constant and effective re-scheduling is an important problem. In this paper, a DAI based resource management system for civil construction companies is presented. To allow effective task allocation, a coalition formation strategy and a specific negotiation algorithm, specially adapted to this class of problems, including intra and inter coalition negotiation is introduced. Finally, to illustrate the proposed solutions, a concrete application example is presented.

Keywords

DAI, Multi-Agent Systems, Negotiation, Coalition Formation

1 INTRODUCTION

Distributed Artificial Intelligence (DAI) is nowadays one of most dynamic AI research areas. In consequence, an increasing number of practical applications have been developed based on DAI techniques with interesting results (Parunak 1995). Air-traffic control (Steeb 1988), power systems management - ARCHON (Wittig 1992), robotics (Oliveira 1991) and particle accelerator control (Jennings 1993) are some examples of problems that have been successfully addressed using DAI based solutions.

In this paper we explore the usefulness of the DAI concepts in the resource management on typical civil construction companies. The inherent geographical distribution over a wide area (usually country or continental size), great resource diversity, easy task decomposition and the multiplicity of functions that the majority of the agents can play are some of the reasons why

this problem is so attractive for the Distributed Artificial Intelligence. Also, the great dynamics due to the large number of possible changes in the initial conditions, forcing the replanning of a great number of tasks, constitutes an exciting challenge. Since we are specially interested in resource management and not in the task planning details, our system's input is a set of tasks to be executed and the pretended output is a group of resources that must be the more appropriated and cost effective to execute it.

For our solution design we opted for a very detailed modelling. Due to the great diversity of physical entities involved and the very common differences between entities belonging to the same class, our choice have been the definition of a software agent for each physical resource containing all it's particularities. In an inherently geographically distributed problem like this, where a company possess many edifications in distant places, it is unrealistic to ignore the communication problems always present in real-world installations all the edifications are intended to have at least a computational equipment where reside the software agents representing the physical entities situated at that local. This way, if there are communication problems and any resource is needed, the system will be able to assign it any entity locally available. If the problem is not urgent, impossible to solve locally or the found solution is not acceptable it will be globally solved as soon as the connection to the rest of system is established. This philosophy has as a consequence the need of a moving capability for all the software agents because they must always follow their physical counterparts. Therefore, our software agents will be spreaded out by a large number of computational equipment in different geographical places. When a new task is launched on the system, it must be broadcasted to all the software agents and the interested agents will bid for it presenting their conditions - costs, production characteristics, etc. In a large system like this, containing a huge number of different software agents, the simple announcement of each task to all the agents can represent an important overhead. In order to reduce this communication overhead we decided to use communication facilitators (Oliveira 1995) that optimise the task announcement by addressing the messages directly to the potentially interested agents.

2 TASK ASSIGNMENT

The task assignment policy will be based on an announcement, bid and negotiation process inspired on the contract net algorithm proposed by Randall Davis (Davis 1983). Several extensions to the contract net have been already presented (Kuhn 1993; Mallone 1988; Sandholm 1995a,b) but none of them including inter and intra-coalitions negotiation. The modifications now introduced to the contract net tend to represent the market oriented behaviour pretended for the agents in this system. It is an objective of the algorithm to reflect the system's market conditions on the calculation of the operation costs of any agent. Therefore, the working cost of any agent must not be based only on factors like operation cost, maintenance and depreciation but it must also reflect other factors like competitiveness, in a market oriented style (Wellman 1994).

In the system, all the agents are exclusively trying to maximise their own benefits. They don't have any global information and they don't take any care about the global profits. How will we form a society of selfish agents that must interact between them to negotiate the execution of the announced tasks and obtain satisfying solutions from the global point of view ? How can we introduce in the negotiation schema issues like the number of available executors for a task

? It seems quite obvious that the execution cost for a task when there are lots of candidates for it, must be lower than it would be if there were just one or two candidates.

First of all let us characterise the tasks we are dealing with. Our system wants to manipulate tasks like excavation, land movement, etc, that are executed by isolated or by groups of machines. It means that if we want to move a large quantity of land it can, for example, be done by a large or by two or more smaller excavators. The costs involved in each of the available options will usually guide the selection process. To accomplish this objective, the system must be able to consider coalitions between agents which are not, wherever alone, powerful enough to execute a specific task but that together can do it. Another important feature of the new algorithm is the ability to reflect on the resulting costs the competition between the agents to perform each task. Time constraints are also a source for interesting problems in this kind of negotiation algorithms. Very often an announcing agent cannot wait indefinitely for the best solution. Even if the adopted solution is not the best solution possible, after a predefined timeout the best one at the moment must be chosen. When this happens, the strategic issues take a special importance. In this situation, the initial proposals are specially important because the time limit doesn't allow the adaptation to the market conditions and probably there will be no chances to modify it. Therefore, a high initial cost can lead to a refusal and an "employment" situation. On the other hand, a low initial proposal can lead to a task sub-valorisation an consequently loose of profits. This is a very interesting research area where we expect to progress further and obtain very important conclusions.

Note that when we are talking about costs, we mean not only the classical working costs but it can be also included other factors like speed and quality of the execution itself.

3 NEGOTIATION ALGORITHM

In our system, each agent must justify his presence by getting some gain in consequence of his activities. This way, when any agent announces a new task, all the possible executors will respond bidding for it and submitting an estimated activity price.

Each agent will estimate the cost for the execution of each task based on the following factors:

- travel cost: in a geographically distributed system the cost of moving a resource between two far places is usually important. So, the price of moving the resources from their actual place to the new working place will constitute one of the items of the estimated price for each task execution.

- depreciation cost: the capital invested on each resource determines that his existence on the system must be paid. This way, any resource has an associated cost just because it exists and if it is not working, it is considered that he loses this amount.

- operation cost: almost all the resources we are interested in have an operation cost resulting from the fuel consumption, tires degradation, engine revisions, etc.

- operator cost.

- profits: in order to compensate the periods of inactivity resulting from "unemployment" or repairs when the agents loose money, they must try to get some profits from their activity.

At the first look, we can imagine that in a system like this all the agents must offer their best price and the best solution is simply to select the lower price (remember that we consider equivalent quality for all the competitors). However, we can adopt a different perspective. In fact, an agent will usually bid with a price resulting from the sum of all the referred factors -

travel, depreciation, operation and profits. However, if the result of asking this price is the unemployment, it would be a better solution to lower the price by reducing his profits. As a consequence, the proposed price must be dependent not only on the agent own conditions but also on the competitors. More generally, any price higher than the agent's operation cost added to the travel cost will be better than to stay inactive. As a result each agent has available a large negotiation margin (see next figure).

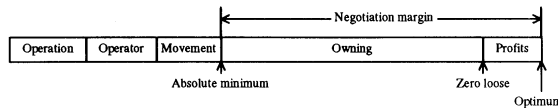


Fig. 1 - Agent's negotiation margin.

To achieve this kind of price adjusting accordingly to the market situation, the system implements an interactive algorithm composed by successive steps.

The algorithm has the following five steps:

- 1st step - announcing - the announcer agent sends the task announcement to all the potentially interested agents.
- 2nd step - task evaluation - the interested agents compute the estimated price for the task execution and send it to the announcer.
- 3rd step - selection phase - the announcer collects the bids, rejects those coming from agents that don't fulfil (from his point of view) basic pre-conditions for election and selects the better price received.
- 4th step - market manipulation - the announcer sends to the non rejected announcers the best price received.
- 5th step - price adjusting - the agents that offered their services receive the lowest verified price and evaluate the possibility of offering a price lower than this one. If it is not possible then the agents retire themselves from the process sending a message to the announcer communicating this fact. If a slightly lower price is reachable then the agent computes a new price and sends it back to the announcer.

These last two steps are repeated until all the agents except one sort out of the process or some timeout arrives. In the later case the best proposal at that time will be chosen.

An important question was intentionally left out of discussion. How to deal with the possibility of coalitions formation [Zlotkin 93; Ketchpel 94] ? How must we handle situations where two or three agents in a coalition can compete with other agents or coalitions of agents ?

This is a realistic situation if we think, for example, about the problem of land dumping. Suppose that we need a land dumping power that can be effectively executed by a two big (and expensive) or by three smaller (and less expensive) dumpers. The negotiation algorithm must have the capabilities to allow this kind of combinations and integrate them on the price adjusting process.

In order to allow coalitions formation, the previous algorithm must be slightly changed introducing a different approach from the one presented in related work like (Zlotkin 1993; Ketchpel 1994).

The 3rd, 4th and 5th steps of the negotiation algorithm must be rewritten and a new step must be added as follows:

- 3rd step - selection phase - the announcer collects the bids and rejects those coming from agents that don't fulfil (from his point of view) basic pre-conditions for election. The

announcer must then calculate the possible coalitions that can solve the problem. This is an exponentially complex problem because with N agents involved we will have $2^N - 1$ coalitions. However, a great number of coalitions can be immediately rejected through the application of some simple constraints. In each one of the approved coalitions one of his members must then be designated as “coordinator”. As we will see, the coordinator will be the agent responsible for the price adjusting within the coalition members. This election will be done trying to distribute evenly the responsibilities by all the agents in the process. Finally, he must send to all coordinators the information that they are coordinators of a coalition, the composition of the coalition team, the individual prices proposed by each of the agents in that coalition and the best price achievable in this stage. In the limit, it will also be possible to have “coalitions” of one single agent.

- 4th step - market manipulation - the announcer sends to the coordinators the best price achievable in this stage.

- 5th step - price adjusting - the agent coordinator of each coalition evaluates the possibility he has of improving the coalition price. For that, he has to establish a negotiation with the coalition partners as he will see later. If it is not possible to obtain a price lower than the best already in the possession of the announcer, the agent retires the coalition from the process sending a message to the announcer communicating this fact. If a lower price is achievable he sends the new offer to the announcer.

- 6th step - price selection - the announcer, after reception of all the answers from all the coordinators (offering better conditions or quitting), evaluates the best offer and communicates the new best to all the active coordinators.

These last two steps are repeated until just one coalition remains active or some timeout is achieved.

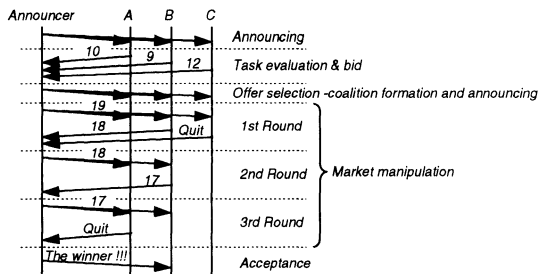


Fig. 2 - Negotiation process example.

The intra-coalition negotiation is also an interesting process. It will be done accordingly to the following algorithm:

- 1st step - the coordinator, after receiving the information about the coalition constitution and the best offer obtained by the announcer, calculates the percentage of the initial cost that the team must decrease in order to obtain a lower price proposal.

- 2nd step - the coordinator sends to all the agents in the coalition (including himself) the lowering percentage.

- 3rd step - the agents respond to the previous message with either an acceptance or a rejection. In the latter case, the agents will inform about their minimal acceptable price.

- 4th step - the coordinator receives the answer from all the agents involved and calculates the new coalition bid. If some of the agents didn't accept the proposed cost reduction and sent a minimum acceptable value, but some other accepted it, a new percentage must be now calculated considering these minimum values. The new decreasing percentage must then be proposed again to the remaining agents (those that accepted the previous percentage) by going back to step 3. If there are no more accepting agents the algorithm steps forward to step 5.
- 5th step - the coordinator informs the announcer about the new coalition position - a better offer or a withdraw.

This process is illustrated in following figure:

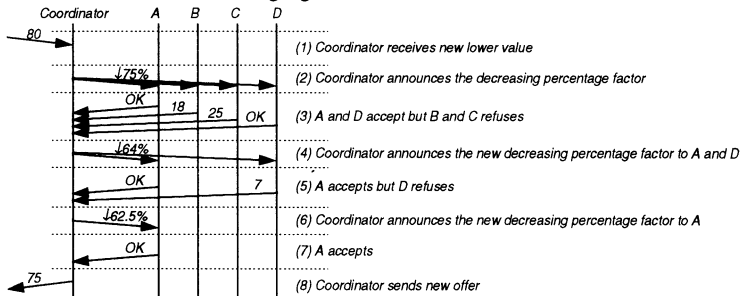


Fig. 3 - Inter-negotiation example.

In the above example, suppose that the initial prices of agents A, B, C and D were respectively 40, 20, 30 and 10. Therefore, the coalition initial value is 100, the sum of the partial values. After having received the information about the coalition composition and partial prices, the coordinator receives the information that the best offer at the moment is 80. Consequently, he calculates that the decreasing percentage must be 75% and proposes it to the coalition members. Unfortunately, only agents A and D accept this reduction. Agents B and C respond with their minimum values of 18 and 25. The rejection of B and C provoke that the coordinator calculates a new percentage (64%) to compensate the difference and proposes this new value to the remaining agents, A and D. This process is repeated again because the rejection of D and finishes when agent A accepts to reduce to 62.5% of its initial value, compensating the rejection of all the other partners. Note that, if all the agents reach their minimum the best possible offer is the sum of all this values. If it is lower then the received value (80 in this example) it is proposed, otherwise the coalition quits.

4 THE SELF DEPRECIATION PROBLEM

The approach we have just presented arises some particular problems. Suppose that an agent simultaneously belongs to two or more competitors coalitions.

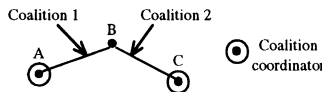


Fig. 4 - The multi-coalition problem.

Accordingly to the last algorithm, agent B will participate on the dispute between coalition 1 and coalition 2. In this way, he will reduce his price whenever it is asked by any of the

coordinators (A and C), until he reaches his lower limit. This is quite unnatural because, if he belongs to both coalitions, he should keep his price and try to force the other agents to argue between them. There are several possibilities to deal with this situation.

4.1 Common elements elimination based on the initial offer

The simplest solution for this problem is the elimination of this kind of situations through an external selection of the partner of the common agent(s). It is possible for the coordinator, when he is forming the coalitions composition. To avoid this problem through the immediate selection of the coalition that has a best initial proposal. This solution is quite simple but can lead to important deviations from the possible best final solution. In fact, the elements that have proposed the initial values aren't always those that can have the best final price. With this strategy, the system can eliminate good coalitions without giving them the opportunity to negotiate.

4.2 Common elements elimination based on direct competition

Another solution is a direct competition between the non-common agents of the coalition in dispute in order to get the right to stay in the negotiation process. With this strategy, agents A and C will compete between them and the best of them will be selected to form the coalition with B. Following this policy, we guarantee that we have chosen the best companion for B. However, if the number of coalitions and crossed coalitions increase this process can be quite complex and fastidious.

4.3 The frozen costs solution

Another elegant solution is what we call the "frozen costs solution". This solution is based on the simple sentence - if we are leading, we have no reason to modify. This way, the leading coalition elements will keep their prices without changes until they are approved or they lose the leading position.

Suppose that we are in the situation illustrated in figure 4 and the proposed costs for A, B and C are 20, 50 and 30 respectively. Coalition 1 presents the best offer (70) and is leading the process. It implies that agents A and B freeze their costs. In this situation, agent C, coordinator of coalition 2, will try to ask agent B to reduce his price in order to achieve a better offer than actual best 70. Because B is cost frozen, he disagrees and keeps the initial value. This way, agent C must reduce the price alone. Suppose that he can reduce to 10. The new coalition 2 value will therefore be 60. Coalition 2 is now leading and agent A must try to beat this new offer. He will then ask B to lower his value but he will deny because he is still in the leading. To overcome that value agent A must reduce to less than 10 what is impossible for him. The negotiation process finishes with the victory of coalition 2 with the value 60. Note that agent B didn't reduce his value along all the process what is exactly what it was pretended because he is considered to be a critical resource that is absolutely needed for the task execution.

A small detail must be pointed out here. What to do in the case of equality between two or more coalitions sharing the best price? If all are designated as leading coalitions there will be

no competition between them and the system can stop without reaching the minimum. If none of them are selected as leading, all of them will be on fighting and the common agent will reduce his price without reason. To solve this problem, it is enough that the announcer randomly selects one of the drawn coalitions as leading. This will give rise to the others to lower their offers. If any of them changes his offer, the system will skip from this situation. If none of them can lower the previous offer, the randomly chosen coalition must be selected as winner because it is the unique that can eventually reduce it's value.

5 APPLICATION EXAMPLE

Let's see how the algorithm works applied to a real world problem. Suppose that we need to move 54000m³ of land in the first 20 days of May, which corresponds to a power of 2700m³/day. This announcement will be received by all the agents potentially interested in the activity (it is all the agents that are able to perform the activity earthmoving). The agents that have potential abilities for that specific activity will then answer with their proposal. Suppose that six agents are available. In the cost calculations a ownership period of 5 years, an estimated usage of 2000 hours/year and a residual value of 35% was used for all the agents (see table 1). The agents will calculate their bids and propose themselves for executing a fraction (or the totality) of the activity accordingly to their own capabilities. The produced offers are shown on the last column of Table 1.

Agent	Model	Delivered Price [Kesc]	Owning Costs [†] [Kesc/Hour]	Operation Costs [†] [Kesc/Hour]	Operator's Costs [†] [Kesc/Hour]	Estimated Production [m ³ /Hour] [‡]	Distance to working place [Km]	Cost [Kesc/Hour] [†]
A	D8	70000	8,33	3,36	3,00	360	300	16,91
B	D6	33000	3,93	2,26	3,00	160	200	11,07
C	D7	54000	6,43	2,66	3,00	280	150	13,67
D	D8	68000	8,09	3,36	3,00	360	250	17,97
E	D7	52000	6,19	2,66	3,00	280	50	13,75
F	D6	31000	3,69	2,26	3,00	160	200	11,24

† - Estimated accordingly to (Caterpillar 1992).

‡ - Estimated production for a 50m dozing distance with universal blades for D8 and D7 and straight blades for D6 at a 80% ideal productivity accordingly to (Caterpillar 1992).

Table 1 - Agents characteristics summary.

Note that in the hourly prices indicated in the last table it is included the moving costs from an agent actual place to the desired working place. In face of this proposals, the announcer will select the coalitions that are in conditions to perform the task (see Table 2).

Coalition	Coalition Team	Initial Cost	Round 1	Round 2	Round 3	Round 4	Round 5
1	A,B	27.97	25.18	-	22.74	X	
2	A,C	30.58	X				
3	A,D	34.88	25.61	X			
4	B,D	29.03	26.13	23.52	-	21.80	X
5	C,D	31.64	26.77	23.63	X		
6	A,E	30.66	X				
7	C,E	27.42	-	24.68	22.43	-	X
8	D,E	31.72	26.85	24.45	X		
9	A,F	28.15	25.33	24.32	22.88	21.97	X
10	B,C,F	35.97	X				
11	D,F	29.21	26.29	23.66	22.74	21.29	-
12	B,E,F	36.05	X				

- the coalition that is leading the negotiation doesn't send any message.

X - message indicating that the coalition retires from the process.

Table 2 - Price evolution along the negotiation process.

All the other coalitions are immediately rejected because they are not enough to solve the problem or they have any redundant element. After this selection process, the announcer must elect the coordinator of each team. These elements are identified in the coalition teams column by the italic bold letter. The next step is the coalition teams and costs announcement to the respective coordinators. Then, the first negotiation round begins and the leading coalition (coalition 7 in this example) is advised to freeze its elements costs. After this, all the coalition coordinators will receive the best offer at the moment (27,42) and try to propose a better bid. For this experiment we decided that all the coalitions lower their prices in 15% steps until they reach their minimum. As we can see in table 2, after the first negotiation round we have a new best value of 25.18. In this first round, coalitions 2, 6, 10 and 12 quit because they can't beat the value 27.42. After five negotiation rounds the solution is found because all but coalition 11 quit. Note that in each round, for obvious reasons, the leader of the last round doesn't propose any new value. The underlined values represent coalitions affected by the frozen costs policy.

6 CONCLUSIONS

An appropriated Multi-Agent System (MAS) architecture for dynamic resource management in the civil construction framework was presented. A coalition formation strategy and a specific protocol description, including intra and inter coalition negotiation were introduced and an application example was developed in order to demonstrate the proposed solutions. Interesting contributions from this work can be related with two different aspects. First, regarding the use of facilitators, a special kind of Agents aiming at reducing communications overheads; second, concerning the flexible Negotiation protocol for both helping on "intra" coalition bid generation as well as for compared evaluation and appropriate selection of the best bids.

As a next step in this MAS for real scenario in the civil construction sector, we intend to develop agents further enhancing them with some learning capabilities.

7 REFERENCES

- Caterpillar (1992). Caterpillar Performance Handbook. 23rd edition. Caterpillar Inc., Peoria, Illinois, U.S.A.
- Davis R., Smith R. G. (1983). Negotiation as a metaphor for distributed problem solving. In *Artificial Intelligence*, 20(1), pages 63-109.
- Jennings, N. R., Varga, L. Z., Aarnts, R. P., Fuchs, J., Skarek, P. (1993). Transforming standalone expert systems into a community of cooperating agents. *International Journal of Engineering Applications of Artificial Intelligence*, 6(4):317-331.
- Malone T. W., Fikes R. E., Howard M. T. (1988). Enterprise: A market-like task scheduler for distributed computing environments. In B. A. Huberman, editor, *The Ecology of Computation*, pages 177-205. North-Holland Publishing Company, Amsterdam.
- Oliveira E., Camacho R., Ramos C. (1991). A multi-agent environment in robotics. *Robotica-Int. Journal of I. E. Research in Robotics and Artificial Intelligence*, Vol 4, Part 9. University Press. Cambridge.
- Oliveira E., Garrido P. (1995). Cognitive Cooperative Facilitators. *Proceedings of IEEE Int. Conf. on System, Man and Cybernetics*, Vancouver.

- [Parunak 95] Applications of distributed artificial intelligence in industry. In O'Hare, G. M. P., Jennings, N. R., editors, *Foundations of Distributed AI*. John Wiley & Sons.
- Ketchpel S. (1994). Forming coalitions in the face of uncertain rewards. In *AAAI*, pages 414-419, Seattle, WA, July 1994.
- Kuhn N., Müller H. J., Müller J. P. (1993). Task Decomposition in Dynamic Agent Societies. In *Proceedings of the Fifth European Workshop on Modelling Autonomous Agents and Multi-Agent Worlds (MAAMAW-93)*, Neuchatel, Switzerland.
- Rosenschein J. R., Zlotkin G. (1994). *Rules of Encounter*. MIT Press.
- Sandholm T. W., Lesser V. R. (1995). On Automated Contracting in Multi-enterprise Manufacturing.
- Sandholm T. W., Lesser V. R. (1995). Issues in automated negotiation and electronic commerce: Extending the contract net framework. In *Proc. First International Conference on Multiagent Systems (ICMAS-95)*, San Francisco, June 1995.
- Steeb, R., Cammarata, S., Hayes-Roth, F. A., Thorndyke, P. W., Wesson, R. B. Distributed intelligence for air fleet control (1988). In Bond, A. H., Gasser, L. editors, *Readings in Artificial Intelligence*, pages 90-101. Morgan Kaufman.
- Zlotkin G., Rosenschein J. S. (1993). One, Two, Many: Coalitions in Multi-Agent Systems. *Proceedings of the Fifth European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, Neuchatel, Switzerland.
- Wellman M. P. (1994). Market-oriented programming: some early lessons. Department of Electrical Engineering and Computer Science, University of Michigan. Ann Arbor.
- Wittig, T., editor (1992). *ARCHON: An Architecture for Multi-Agent Systems*. Ellis Horwood.
- Wooldridge, M. J., Jennings, N. R. (1994). *Intelligent Agents: Theory and Practice*.

8 BIOGRAPHY

Eugenio Oliveira is Associated Professor at the University of Porto. He is in charge of Artificial Intelligence & Robotics Group (NIA&R). After graduating in Electrotechnical Engineering (72) he was R&D Engineer at BBC (now ABB) in Switzerland (72-74). He was Assistant Lecturer in Porto (75-80), researcher in UNLisbon (80-84) where he got his PhD in AI, guest scientist at IBM IEC in La Hulpe (84-85) and, since then, Professor at the University of Porto. Main topics of interest in his group are Multi-Agent Systems (Negotiation, Cooperation, Learning, Belief Revision) and its applications as well as Intelligent Robotics.

Eng. José Manuel Fonseca received his BSc on Electronics and Telecommunications from the Polytechnic Institute of Lisbon, the Computer Engineering Degree and the MSc Degree from the New University of Lisbon (UNL). He is currently Assistant and PhD candidate at the Department of Electrical Engineering of UNL. His main interests are: Multi-Agent Systems, Distributed Artificial Intelligence and Machine Learning.

Adolfo Steiger-Garção is full professor at Department of Electrical Engineering of UNL. He is also President of UNINOVA Institute and director of the Intelligent Robotics Center. His main research interests are Intelligent Robotics and Artificial Intelligence. He is responsible for several international projects and has more than 100 publications on national and international conferences.