

An artificial maieutic approach for eliciting experts' knowledge in multi-agent simulations

François Sempé¹, Minh Nguyen-Duc^{1,2}, Stanislas Boissau³, Alain Boucher¹,
Alexis Drogoul⁴

¹Institut de la Francophonie pour l'Informatique (IFI), Hanoi, Vietnam
francois@ifi.edu.vn alain.boucher@auf.org

²Laboratoire d'Informatique de l'université de Paris 6 (LIP6), Paris, France
minh.nguyen-duc@lip6.fr

³Wageningen University, Netherlands, and Ecole des Hautes Etudes en Sciences
Sociales, Paris France
sboissau@fpt.vn

⁴Institut de Recherche pour le Développement (IRD), Bondy, France
alexis.drogoul@bondy.ird.fr

Abstract. Models of human behaviours used in multi-agent simulations are limited by the ability of introspection of the social actors: some of their knowledge (reflexes, habits, non-formalized expertise) cannot be extracted through interviews. The use of computer-mediated role playing games put these actors into a situated stance where the recording of their "live" behaviours is possible. But cognitive processes and motivations still have to be interpreted.

In this paper, we propose an artificial maieutic approach to extract such pieces of knowledge, by helping the actors to better understand, and sometimes formulate, their own behaviours. The actors are playing their own roles in an agent-mediated simulation and interact with agents that question their behaviours. The actor's reactions and understanding are stimulated by these interactions, and this situation allows in many cases to reveal hidden knowledge. We present here the first results using two complementary works in social simulations, one in the domain of air traffic control and one in the domain of common-pool resources sharing.

1 Introduction

Predicting the evolution of a social organisation in which a number of people are involved in collective (collaborative or concurrent) activities requires to undertake simulations. One of their most common applications is to evaluate the outcomes of new management policies or working procedures before they are applied, especially in critical domains where security is essential, like air traffic control. In this respect, agent-based techniques are broadly used to simulate human beings thanks to the inherent "social" nature of agents, which enables modellers to easily represent in an artificial environment processes like interaction, communication, or collaboration between people.

In order to create agents that represent human beings, however, it is necessary to model relevant parts of human knowledge into sets of behaviours, decision rules, or heuristics which will be available for each agent. When the knowledge is already available through previous (sociological, anthropological, ethnological) studies, the translation into a computational model, although it can take some time, is mostly a matter of finding an appropriate supporting architecture. When it is not available, it requires to be extracted from experts or directly from actors of the target human organisation, through interviews, inquiries, or field experiments. However, most extraction methods encounter a number of difficulties when it comes to model informal knowledge that come from the experience of the actors rather than through classical learning.

We propose to tackle this issue by using simulation as a support for building such models, thanks to a computer mediated, continuous dialogue between human actors and artificial agents [6]. Our hypothesis is that the actor can more easily describe his behaviour or knowledge when placed in situation in a role-playing game supported by a simulation. In what we call an artificial maieutic approach, the agent questions the actor and tests his reactions, either directly (“Why such an action?”) or indirectly (through a modification of the perceptions available) in order to explore his informal knowledge.

Two experiments are presented in this paper in order to illustrate our approach, the first one in the domain of air traffic control and the second one in social sciences.

2 Traditional Social Simulation

The concept of “multi-agent system”, since its emergence in the 1980s, has always been considered, among other things, as an interesting modelling and simulation tool for social sciences. In the 1990s, if we set aside “toy simulations” serving as illustrations of social theories (like Sugarscape for instance), social multi-agent simulation began to be used in critical domains like military [8] or industrial research (a good example is the simulation of a large population of consumers made by France Telecom in [13]).

More recently, and in a similar way to what we propose in this paper, researchers from social and agronomic sciences working on the management of renewable resources have explored agent-based approaches to model and understand the outcome of, for instance, different sharing policies on the availability of resources [2]. Their goal is to use multi-agent simulations, simultaneously as a support for experimental research and as a computer-aided training and decision-making tool for actors.

3 Artificial Maieutics for Eliciting Informal Knowledge

In order to make such simulation, it is necessary to model human behaviour. Usually, the model is built through an iterative process with the help of interviews between a modelling expert (the one who builds the model) and some social actors. This method is limited by the ability of actors to describe and explain their actions. But a part of the knowledge to be elicited from actors is not accessible through interviews or more general inquiries (sociological or anthropological). Reflexes,

habits, reactions to unexpected situations or behaviours refined by experience represent some informal knowledge hard to capture and to formalize. How can we have access to this kind of knowledge in order to improve the models?

In this paper, we got inspiration from *human-in-the-loop* experiments which directly integrate human actors in the running loop of a simulation with the help of a dedicated interface. Our approach relies on the following assumption: immersed in a (simulated) real-conditions situation where he is asked to play his own role, the actor can more naturally exhibit these informal behaviours, and hopefully better understand and describe them to the modellers. In turn, using the simulation as a support enables the modellers to be more accurate when trying to refine the behaviours programmed: as a matter of fact, the questions they ask to the actor are grounded in real situations and they can ask them as soon as this situation appears within the simulation.

We call this approach “artificial maieutics” in relation to the questioning method used by Socrates (Greek philosopher of the 5th century B.C.) to make his interlocutor discover by himself some non-conscious knowledge. In artificial maieutics, the role of the questioner can be played either by an expert or, more interestingly, by an artificial agent during the simulation. In other words, an agent can be attached to the interface as an assistant, which role is to interrogate the actor about his actions, and to record and use the answers in order to improve the model. These two methods share the same basis (illustrated on Figure 1): The first step is to develop a role-playing game where the social actors are asked to play their role in carefully chosen scenarios; this should result in a set of logs or traces, which, in the second step, will be used (either manually or automatically) to program artificial agents.

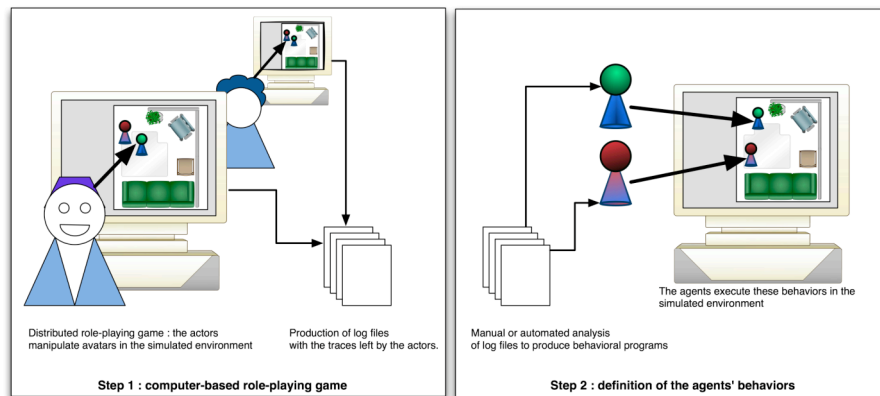


Figure 1 – The first two steps of artificial maieutic experiments

Once the simulation (even with minimal behaviours) is up and running, it can be used as a basis for the maieutic step. Two methods are then available: one which involves active discussions between the actor and an expert, the other active interactions between the actor and “his/her” agent. The first one is usually used when the domain is not formalized or when the goal of the simulation is not sufficiently defined. The second strongly relies on the knowledge available about the domain: the agents have to be aware of the goal of the simulation to understand the answers of the

actors. These two methods (see Figure 2) are not antinomial and can be used in sequence (or in parallel) to refine the behaviours of the agents and build a better model.

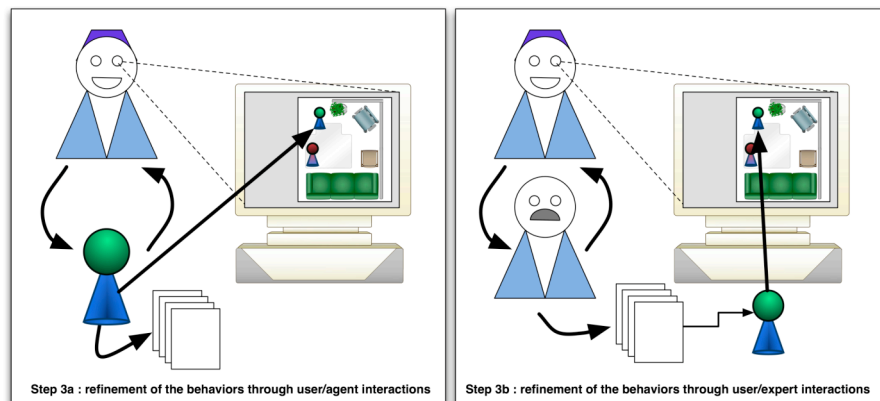


Figure 2 – The two different methods of artificial maieutics

In order to illustrate the use of these two methods, we present two experiments. The first one shows the use of the user/expert interactions : a group of students participate in a role playing game about common-pool resources sharing and are asked to build and discuss with their teachers a model of their own behaviours.

The second one describes how interface agents can be used for questioning air traffic controllers engaged in a simulation of their daily professional activity (these experiments have been used to validate innovative organizational methods among controllers [11]).

4 Artificial Maieutics and self-modelling

In this experiment, some computer science students were asked to make a model of their own behaviour during a role playing game. We present here the first step of these experiments where students of the French-speaking institute of computer science (IFI - Institut de la Francophonie pour l'Informatique) in Hanoi, Vietnam, have no artificial maieutic tools to complete the modelling task. This experiment has two purposes. First, it will allow us to better understand the difficulties that someone has to face when making a model of himself: we expect to learn more on how to design these tools. Second, this experiment will be used later as a reference for an evaluation of the artificial maieutic tools.

4.1 Experiment Description

The Settings. The IFI students who took part in this experiment are graduated computer engineers (5 years of university). Some of them are teaching at the university or working in a company. Their abilities in computer science allow them to

process the whole modelling task, until the coding of an agent that reproduces their own behaviours.

The modelling workshop was divided into 5 steps that took place during a week:

- 1) Explanation of the game and play a game.
- 2) Just after the end of the game, students were asked to describe their behaviour without any help.
- 3) The day after step 2, the students corrected their first descriptions with the help of a tool to replay their game.
- 4) From the previous written description, a computational model was built through the writing of pseudo-code and code for an agent that model their behaviour during the game.
- 5) Finally, the students had to evaluate their model comparing the real game and a simulation of the game running with the agents they coded.

The Game of Friends. The students took part in a role playing game which is similar to another one played by farmers from northern Vietnam (the game of Buffalos) as part of political science research. This research program is focused on people's behaviour when they face the growing scarcity of common-pool resources [3]. In this paper, we are only interested in modelling questions and we will not tackle the social side of the experiment.

The rules of the two games are the same, only the scenario is changed. The game of Buffalos does not suit student players because they are not familiar with the life of a farmer. Unfamiliar situation could lead to fanciful behaviours.

At the beginning of a game, one player starts with a certain amount of friends. At every loop, he has to download a movie on internet (which is against laws of copyright and we do not recommend) for each friend in order to keep him satisfied. During the game, the total number of possible downloaded movies decreases. A friend may not receive his movie, which turns him angry. Angry friends may leave the player. At each loop, the player can make new friend (spending "free time" currency) or leave some old friends. Download resources being shared by all players and the number of friends of one player influence the state of the whole system.

The game has been implemented as client/server application. One client is an interface for a human player or an agent that acts like a human player. Thus it is possible to mix agent players and human players in the same game.

One of the originality of this game lies in the lack of goal: players are free to choose their own goal, there is no winner nor loser. At contrary, most economical games have explicit goals like the maximisation of a profit [12]. In other respects, the student taking part in the game of Friends received no information about it before the beginning of the workshop. Thus, they had no means to prepare themselves in any way, for instance, in building a strategy that would be easy to implement into an agent. The lack of goal for the game and information about the workshop serves the purpose to avoid ad hoc strategies.

23 students took part in the experiments. One game is played by 5 players, humans and artificial and lasts 25 loops. 2 games have artificial players, without the knowledge of human ones. All games took place at the same time, in the same room, but players did not know with whom there were playing.

4.2 Results

Game's Dynamics. First, let us notice the existence of 2 steps during a game. From the beginning to about the sixth loop, resources are in excess and all friends can be satisfied. From the sixth turn, resources were getting scarce, and the game entered in a step of shortage. Then, depending on players actions, the shortage stay high (Figure 2) or the number of resources and the number of friends reach an approximate equilibrium.

The game of friends is thus characterized by crises (shortage) and the need for the players to adapt their behaviours.

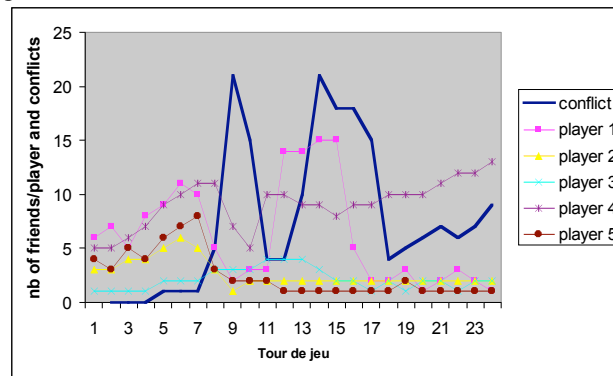


Figure 3. This chart shows the evolution of the number of friends for each player during 2 games. In this game, the « selfish » behaviour of one player – keeping a lot of friends - maintains the number of conflicts (i.e. the number of unsatisfied friends) at a high level.

Modelling Effect of Coding. In written descriptions of the students' behaviours, we found two opposite faults: the lack of precision ("when conflicts for resources are few...") and the lack of synthesis ("on loop n° 4, I made a new friend..."). Forgetting occurs too: one explains when he makes new friends but tells nothing on the conditions of quitting a friend. In other respects, a lot of description refers to random. "I choose internet sites at random", "I quit friends at random"... Well, in most of the cases, they did not act at random at all as they say. It is enough to watch the game log to be convinced for that. The extensive use of the expression "at random" has the purpose to cover the ignorance of a player who cannot explain why he acts in a certain way. As students had no time to prepare a strategy before the game starts, non-conscious behaviours - hard to describe - have emerged.

The written description was followed by the writing of the pseudo-code for the agent that reproduced their behaviour in the simulation. As pseudo-code contains the logic of a program (tests and loops), this step stands for a hidden modelling process. It has had a clarifying effect: students were forced to transform somehow their imprecise written description into a computational model of their behaviour. For instance, the condition "when resources are getting scarce..." become « when there less than 2 unused resources. This transformation is crucial to the self-modelling process. Let us focus on it.

Creativity in Modelling. It appears that the modelling process use both recollection and creativity. As we explained before, the scenario of the game of Friends has 2 steps. For most of the players this configuration has induced a behaviour switch: the number of friends is increased during the first step when resources are abundant, and reduced during the scarce period. What conditions were chosen to trigger the switch?

In one third of the cases, students chose a condition based on the state of their friends, for instance “if one of my friend was unsatisfied during the last turn, then I do not make new friends”. It is a plausible condition as the player could consciously had this perception during the game. But in most of the cases, students had to use their imagination. The one who writes “If more than 80% of internet sites are overused...” did not count the sites during the game, but create a condition that could match his behaviour. One of the students describes explicitly how the recollection of a feeling is turned into model: “I remember getting bothered when resources became rare and I had to change my strategy. A statistical analysis of logged data allows me to discover when I have modified my behaviour, i.e. when the number of resources is less than the total number of friends.”

Tendency to Idealization. As they discover some weaknesses or contradictions in their strategy during the game, a lot of students have the tendency to “improve” their behaviour in the model. They want a consistent behaviour for their agent, although we were repeating all time long that in the game of Friends there are no good or bad behaviour. This idealization can be noticed in such commentary: “I could not play like I intended to because I had very few friends at the beginning of the game” – which is obviously hindsight thought.

Tendency to idealization of behaviour is very problematic as it is judgement itself that is biased. “Is this or not a good and useful model?” becomes “Is it or not a nice model of myself” The students did not want to look stupid in this image of themselves, as anybody else would.

During this workshop, students both had to deal with easy-to-model behaviour (“I dot not like to have angry friends so I quit them”) and hard-to-model ones (“I have been bothered...”, “I played at random...”). Some of those non-conscious behaviours could have been rediscovered through game log analysis, but not all of them, and even when possible, students did not necessarily understand the reason of a given action. Analysis is not enough for understanding. In addition, tendency to idealization of behaviour may bias the whole self-modelling process.

5 A Simulation Tool for Air Traffic Controllers

The second experiment takes place in a specialized domain, where the interactions between social actors are most of the time strictly formalized. Our aim is to show that, thanks to this formalization, artificial agents can be advantageously used for extracting knowledge about the behaviours of the actors in situations impossible to create in real-life experiments. Our tool is based on a multi-agent simulator already implemented for ATC by EuroControl, in which interface agents and functionalities for modelling human behaviour were added.

5.1 Overview of Air Traffic Control

Current Air traffic Management (ATM) system is airspace-based. The airspace is divided in several sectors, the size of which depends on the number of aircraft in the region and the geometry of air routes. There are usually two air traffic controllers to handle the traffic in each air sector: a planning controller and an executing controller. The planning controller works at a strategic level to minimize the number of conflicts or their complexities. The executing controller works at a tactical level to ensure that there are no conflicts, *i.e.* infringements of standard separation between aircrafts by giving instructions to the pilots.

ATM still comprises a higher level of management, *i.e.* the traffic flow management. The flow managers are located in the Central Flow Management Unit and in each control centre. Their task is to (re-)plan the flights at the multi-sector level with as major objective to avoid the congestion and the overload of the controllers (due to the big number of aircrafts to be controlled).

5.2 Agent/Human Hybrid Simulation for an Operational Procedure

A major concern in leaving some loose end to ATM rules is the occurrence of uncontrolled traffic peaks at the entry of a congested area. This phenomenon, often caused by some aircraft “in bunch”, is known in the operational world as “traffic bunching” effect. A way to solve the problem is to structure and organise the arrival flows in real-time. A possible technique is the readjustment of the arrival time of some aircraft at a congested point, thus enabling to “de-bunch” problematical delivery. This technique should enable several controllers and flow managers to collaborate on the traffic for “smoothing” the bunching peaks before they affect the congested area. In our vision, this working group is a kind of social organisation that we can model and simulate.

The investigations undertaken by EuroControl (a European research center in the field of air control) on this collaborative operational procedure need simulation tools able to validate new team-working methods and to be used as support of demonstration bound for real air traffic controllers and traffic flow managers. For this goal, we are implementing an agent/human hybrid simulator in which the human actors (controllers or flow managers) and their assistant agents, are working together like team-mates. We model the interaction between the artificial agents by using STEAM (Shell for TEAMwork), a generic teamwork model described in Tambe et al. [10] The participation of domain experts (*i.e.* controller or manager of flow) is regarded as new for the multi-agent simulations [7] but human-in-the-loop experiments are already largely used at EuroControl for a long time. This participation makes it possible to experiment the maieutic approach of human behaviour modelling.

5.3 Agent/Expert Dialogue

We added to the user interface dedicated to each participant expert an interface

agent playing the role of assistant. This last one can play alone the role of the corresponding expert or only assist him. An expert and his assistant constitute, with respect to the other players, one and only one expert/assistant player. The expert plays the role, which was assigned to him within the simulation. The assistant observes, then proposes some behaviours which can be amended by the expert, these modifications being taken into account by the assistant as well as the results of his observations. This agent/expert relation leads to a dialogue in which the assistant questions “why don't you do that for this reason?” and the expert answers by modifying the behaviour suggested “I modify your proposal because of these entities and for this reason”. The answer helps to improve the existing human behaviour model. This improvement can be made either by hand by the designer after having studied the log of the simulation game or automatically by agents. However the suitable learning techniques remain to be discovered so in this paper we tackle only the first possibility.

5.4 An Air Traffic Control Simulation Game

To go more into the details of some maieutic dialogues already carried out, we describe here a particular simulation game. We define, in agreement with the experts, a coordination protocol between flow managers in order to create the simulation game. We suppose an initial situation like the following one: a flow manager called “requestor” detects a risk of “traffic bunching” about an hour before it affect a congested sector; this risk is caused by aircraft flying “in bunch” which will cross successively the airspace zones managed by other managers called “suppliers”; the “requestor” informs these “suppliers” of the risk and starts a session of common tactic establishment. The two roles defined here are not exclusive, *i.e.* a flow manager can be at the same time “requestor” and “supplier”.

The defined coordination protocol is as the following:

- 1) The “requestor” builds a pre-tactic to solve this “traffic bunching” risk. This pre-tactic is divided into several measures, each of which is dedicated to handling of one of the aircraft flying “in bunch” and managed by a “supplier”. Then it diffuses this pre-tactic to all the “suppliers”.
- 2) Each “supplier” accepts, refuses or modifies associated measures, then diffuses his ideas to the “requestor” and to all the other “suppliers”.
- 3) After having received all the ideas, the “requestor” sees whether there is a refusal. If yes, this coordination failed; if no, it updates and then validates the final common tactics.

The behaviours of the actors presented above are modelled by the different agents plunged in a similar situation (in the context of pre-established scenarii). The goal is to provide to the experts an intuitive vision of these coordination protocols, and to ensure that they can act easily to modify them, in particular on the following points:

- 1) **action**: all actions the managers can do in the protocol, *e.g.*, build a pre-tactic, diffuse the pre-tactic, accept a measure, refuse a measure, modify a measure, diffuse the ideas and validate the final tactics.
- 2) **perception**: all information and data which the managers have to perceive in

order to make decisions, *e.g.*, the number of aircrafts present in each sector, geographical information about the sectors and beacons.

- 3) **reason to act**: the reason for which a manager chooses a specific action and not others, or the reason for which it refuses a measure.

An interaction session between an assistant and an expert player can be described as follows (Figure 4):

- 1) At one time, the expert has to decide to accept, to refuse or modify a measure, *e.g.*, the re-routing of aircraft MSK20N with the new route segment from beacon RBT to beacon LFMN.
- 2) The assistant suggests the **action** “modify the measure” which changes the new route segment, *e.g.* it adds beacon ALBET between two beacons AMFOU and ARMUS by explaining that the controllers of sector LFEUF1 will be “very loaded” while those of sector LFFUJ1 will be less “loaded” (Figure 4a).
- 3) The expert amends this **action** by indicating another new route segment, *e.g.* it takes again the direct way path from AMFOU to ARMUS and removes beacon SINRA.
- 4) The agent asks for **reasons to act** by presenting a question “Why do you modify my suggestion?”
- 5) The expert answers this question by describing his **perception**. He chooses the controllers, aircraft or flow managers who cause the amended action and by specifying a reason for each selected controller, aircraft or flow manager. For example, the expert chooses the aircraft MSK20N as the principal cause, and the specified reason is that this aircraft is already “too” delayed (more than one hour) compared to its initial flight plan and that its route cannot be more lengthened (Figure 4b).

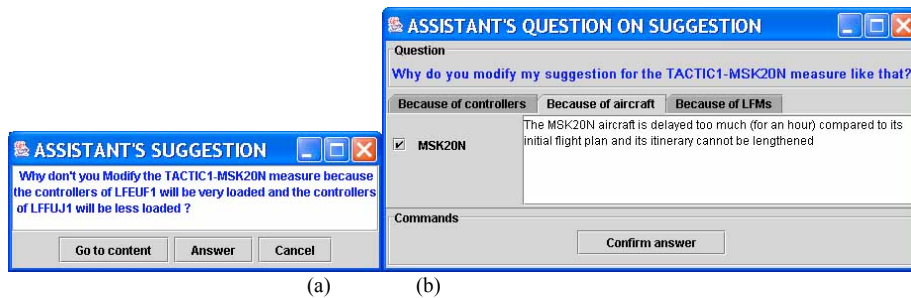


Figure 4. Examples taken from a session of interaction between an assistant and an ATC actor. (a) Suggestion of the assistant with justification. (b) Question of the assistant on the amendment and answer of the expert.

5.5 Example of Extracted Behaviour

In interaction with the expert, the assistant seeks to build and structure a log of “Actions” being based on the **actions** validated or amended by the expert. This log is used as a data source for the design of the simulator, both in a manual mode, because its structure is designed to help the designer formalizing the behaviours of the agents,

and in an automatic mode, with objective to allow the agents to learn by themselves the pertinent behaviours. We present below the structure of an “Action” in the log:

By applying this model at the time of the interaction session described above, the assistant will thus have conserved an “Action” with the following parameters:

Action = (*Predicted_Situation*, *assistant*, *Suggested_Action_Type*, *Suggested_Action*, *Causes_suggestion*, *Reason_suggestion*, *Amended_Action_Type*, *Amended_Action*, *Causes_amendment*, *Reason_amendment*)

The designer would thus be based on the initial flight plans, on the geography and on the “Actions” conserved in the log to formalize agent behaviours. For example, he can add the following abstract rules:

- 1) If the difference between the maximum control capacity of a pair of controllers and the number of aircrafts present in their sector is lower than 3, these controllers will be considered as “very loaded”.
- 2) If an aircraft is delayed for more than one hour compared to its initial flight plan, it will be considered as “too delayed”.
- 3) The route of a “too delayed” aircraft cannot be more lengthened even if the lengthening of the route allows it to avoid the “very loaded” controllers.

And new “Rules of deduction” are created from these abstract rules.

6 Discussion and conclusion

In the domain of political and social sciences, role games and multi-agent simulation have various goals, for example the study or validation of models [2], or the support for negotiation between actors [5]. The usefulness of assistant agents has already been stress, for instance in [4] where the authors want to identify the regularity of interactions between actors.

In this paper, role playing games and multi-agent simulation have been used to propose a new scheme for knowledge acquisition from a human actor to an artificial agent. In other words, this represent a different way of the old questions of how can we give knowledge to an agent and how does the agent can reason using this knowledge? We state in these simulations that we cannot give in one shot all the necessary knowledge to an agent, but only a part of it. This remaining knowledge, coming mainly from human experience, and human instinct, cannot be given so easily. Instead of giving the knowledge straight to the agent, we are giving tools to the agents to observe and step-by-step learn by assisting the human. Instead of resulting in an intelligent systems ready-to-work, the system “only” intend to observe and learn, without short term autonomous actions

In this paper, two complementary works about artificial maieutic were presented. First, in the air traffic simulation, the assistant agent has the possibility to question its own model with the decisions taken by the expert. The questions reflect missing parts in the agent’s model compare to the expert’s model. This missing part has to be filled in some ways, or by internal update from the agent itself or from update from the actor.

The game of Friends experiment is more prospective. We have shown how difficult it is for an actor to understand and explain even simple actions (making new friends or quitting old ones) made few hours ago. In addition, actors re-create a picture of themselves during the modelling process for two reasons: first, because they do not always remember the motivations nor even the conditions of their actions, second because they want to build a “nice” model rather than a true or useful one. If a model is always a creation, it should be as close as possible to reality. How an assistant agent could help an actor to stay faithful to his real behaviour? Explicit questioning, like in the air traffic control application, could be improved in taking advantage of the flexibility of a computer interface. With a simulation, it is easy to filter the perceptions of an actor, modifying actor’s interface. In place of a direct question like “why did you make a new friend”, which may be difficult to answer as we seen, an indirect question could be “what would you do in such situation?”. For instance, in order to establish if one player’s behaviour is driven by the states of his friends or the global state of the system, an assistant agent can test the reaction of the actor when some information is hidden. The agent and the actor can then better state the conditions for a given behaviour. Experimentations on similar situations than the game of Friends will allow us to test the artificial maieutic approach through perception filtering.

References

- [1] P. d'Aquino, O. Barreteau, M. Etienne, S. Boissau, S. Aubert, F. Bousquet, C. Le Page & W. Dare, “The Role Playing Games in an ABM participatory modeling process: outcomes from five different experiments carried out in the last five years”, in *Comm. to IEMSS*, Lugano, June 24th-27, 2002.
- [2] O. Barreteau & F. Bousquet, “Role-playing games for opening the black box of multi-agent systems: method and lessons of its application to Senegal River Valley irrigated systems” in *Journal of Artificial Societies and Social Simulation*, 2001.
- [3] S. Boisseau, “Co-evolution of a research question and methodological development : an example of companion modeling in northern Vietnam” in F. Bousquet & G. Trebil (eds.) *Companion modeling and multi-agent systems for integrated natural resources management in Asia* – In Press.
- [4] A. Candea, H. Hu, L. Iocchi, D. Nardi & M. Piaggio, “Coordination in multi-agent RoboCup teams,” in *Robotic and Autonomous Systems*, vol. 36, 2001, pp. 67-86.
- [5] A. Drogoul, B. Corbara & D. Fresneau, “MANTA: New experimental results on the emergence of (artificial) ant societies,” in *From reaction to cognition*, lecture notes in AI n° 957, C. Castelfranchi & J.P. Müller, Ed. Berlin-Heidelberg: Springer-Verlag, 1995, pp. 13-27.
- [6] A. Drogoul, T. Meurisse & D. Vanbergue « Multi-Agent based simulations : Where are the agents ? » in Sichman J.S., Bousquet, F., Davidsson, P., eds: *Multi-Agent Based Simulation, Third International Workshop*, Lecture Notes in Computer Science, vol. 2581, pp. 1-15, Springer, 2002.
- [7] P. Guyot & A. Drogoul, , “Designing multi-agent based participatory simulations,” in *Proceedings of the 5th Workshop on Agent Based Simulations*, Lisbon, 2004.
- [8] R. W. Hill, J. Chen, J. Gratch, P. Rosenbloom & M. Tambe, “Intelligent Agents for the Synthetic Battlefield: A Company of Rotary-Wing Aircraft,” in *Proceedings of the 9th*

Conference on the Innovative Applications of Artificial Intelligence, Menlo Park, 1997, pp. 1006–1012.

- [9] T. Ishida, “Q: A Scenario Description Language for Interactive Agents”, in *IEEE Computer*, 2002.
- [10] R. Nair, M. Tambe & S. Marsella, « Role allocation and reallocation in multi-agent teams: Towards a practical analysis », in *Proceedings of the 2nd Int. Joint conf. on agents and multi-agent systems* (AAMAS’03), 2003.
- [11] M. Nguyen-Duc, V. Duong & A. Drogoul, “Agent-based modeling and experimentation for Real-time Collaborative Decision-Making in Air Traffic Management”, in *Proceedings of the 24th Congress of the Int. Council of the Aeronautical Sciences* (ICAS’04), Yokohama, 2004.
- [12] E. Ostrom, “Coping with the tragedy of commons”, *Annual Review of Political Science*, vol 2, pp. 493-535, 1999.
- [13] L. B. Said, T. Bouron & A. Drogoul, “Agent-based interaction analysis of consumer behavior,” in *Proceedings of Autonomous Agents and Multiagent Systems* (AAMAS’02), Bologna, 2002.