MASCARET: Pedagogical Multi-Agents System for Virtual Environment for Training.

Cédric Buche, Ronan Querrec, Pierre De Loor, Pierre Chevaillier
Laboratoire d'Ingénierie Informatique
Ecole Nationale d'Ingénieurs de Brest
F – 29 608 Brest
FRANCE
[buche, querrec, deloor, chevaillier]@enib.fr

Abstract

This study concerns virtual environments for training in operational conditions. The principal developed idea is that these environments are heterogeneous and open multi-agent systems. The MASCARET model is proposed to organize the interactions between agents and to provide them reactive, cognitive and social abilities to simulate the physical and social environment. The physical environment represents, in a realistic way, the phenomena that learners and teachers have to take into account. The social environment is simulated by agents executing collaborative and adaptive tasks. These agents realize, in team, procedures that they have to adapt to the environment. The users participate to the training environment through their avatar. In this article, we explain how we integrated in MASCARET models necessary to the creation of Intelligent Tutoring System. We notably incorporate pedagogical strategies and pedagogical actions. We present pedagogical agents. To validate our model, the SECUREVI application for fire-fighters training is developed.

1. Introduction

This study concerns virtual environments for training in operational conditions. We want to simulate and to immerge the learners in their professional environment. This enables them to manipulate the environment so that they can *"learn while doing"*. This idea is driven by the "constructivism" paradigm defined by Piaget [1] and can find a good implementation in virtual reality techniques as presented by [2]. Our definition of Virtual Reality is the one proposed by [3] which proposes to give autonomy to models involving in the virtual environment by giving them the "triple mediation of senses, decision and action". So, the main developed idea is that virtual environments for training are heterogeneous and open multi-agent systems. Those Multi-Agents Systems (MAS) has been presented by [4] using the *VOWELS* model considering a MAS with four vowels: **A**gent, **E**nvironment, **I**nteraction and **O**rganisation. It also has been use for collaborative work simulation by Clancey [5]. From our point of view, we consider the user of a virtual environment as other autonomous agents because he can interact with the environment and with other agents or users in the same way. Then, as [3], we propose to add a last vowel, the letter **U** for **U**ser, in the *VOWELS* model.

Our goal is to provide model to create [6] Virtual Environment for Training (VET) [7] and to integrate an Intelligent Tutoring System (ITS) [8, 9]. ITS aims at providing students with dedicated tutoring. Its goal is to communicate knowledge effectively. It can be composed of fourth components [10]. First, the domain model is the representation of the teacher's expertise [11]. Second, the learner model allows knowing at each instant the state of the student's knowledge and curriculum [12]. Then, the pedagogical model refers to the methods of instruction. Using student's behaviour and his model, the pedagogical model allows selecting learning strategies. Finally, the interface model permits the interaction between the system and the student.

Our objective is to place learners in operational positions in their simulated physical and social environment. First, we show the *MASCARET* model, proposed to organize the interactions between agents and to provide them reactive, cognitive and social abilities to simulate the physical and social environment. It allows defining the fourth models of an ITS. We propose a pedagogical behaviour based on interactions between such models. Then, we present the *SECUREVI* application. Its goal is to train fire-fighters officers to operational management and command. Finally, we conclude and envisage futures work.

2. The *MASCARET* model

Our goal is to train teams to collaborative and procedural work in a physical environment. In this case, we have to simulate in a *realistic* way this physical environment and the *collaborative* and *adaptive* team member's behaviour in the social environment. Evolution of those environments results from simulation of autonomous agent's local behaviour and their interactions. We propose a model, called *MASCARET*, where we use multi-agents systems to simulate realistic, collaborative and adaptive environments for training. This model aims at organizing the interactions between agents and provides them abilities to evolve in this context. In addition, it allows the establishment of models necessary to the creation of Intelligent Tutoring System. *MASCARET* also permits to define pedagogical activities.

2.1. The organisational model

As the users has to be integrated both in the social environment (member of a particular team) and in the physical environment (to undergo a lick of gas for example), we propose, first a generic organisational model allowing to represent the physical and the social environment. The model we propose is founded upon the concepts of `organisation`, `roles`, `behavioural features` and `agents` (Figure 1). [13] has already proposed an organisational model for multiagents systems, but this model, dedicated to the collaborative realisation of procedures, is not enough generic to solve our problem. [14] has also proposed such a model called Agent/Group/Role, but this model seems to be more a pattern for conception than a model which formalizes the concepts of organisation and roles. In our model, the aim of the organisation is to structure interactions between agents; it enables each agent to know its partners and the role they are playing in the collaboration. The concept of role represents the responsibilities (`behavioral features`) played by agents in the organisation. Agents have then an organisational behaviour which permits them to play or abandon a role in an organisation. This behaviour enables also agents to take into account the existence of other members.

This model is a generic model in the way that all the resulting classes are abstract. The organisational model is then derived to implement two concretes organisations representing physical and social environment that have to be simulated in the virtual environment for training.

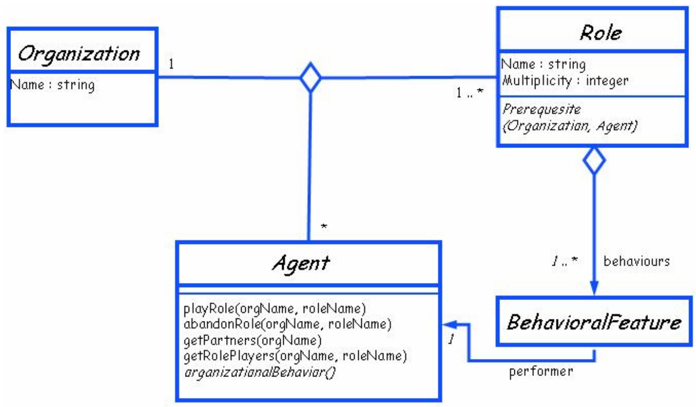

Figure 1. The organisational model.

2.2. The context of simulation

The physical environment.

In a Virtual Environment for Training, the user's (learner and teacher) physical environment must be realistic, interactive and act in "real-time". Then, to reach the constraints of virtual reality, models we use to simulate physical phenomena are obviously simplified. Therefore, the teacher may wants, for pedagogical reasons, inhibits some phenomena. For that, we must propose models which are compatible to a disconnection between the phenomena. Moreover, although all interactions have potentially effects on the two agents involved, we introduced the notion of interaction source and target. We consider then that the interactions between agents have a privileged direction.

Then, the reactive agents' behaviour evolving in physical environment is to perceive situations where there are interactions and to act consequently. A practical limit of the individual based models is that each agent can potentially perceive all others. The complexity of the algorithm is in this case $O(n^2)$. Then, we have to design rules to organize theses interactions between reactive agents. For that, we use then the generic organisational model we have proposed before. The organisation is, in this case, a network where agents are connected together when they are in interaction. We call this organisation an interaction network (`InteractionNet`, Figure 2). To represent the concept of privileged direction in interactions, we define two particular roles called source and target. The goal of source agents is to give information on their internal states to other agents (targets) so that they can compute the interaction's value and their internal state. The interaction can be detected by the two agents involved, but, for "real-time" computation reasons, it is better if only one agent detect it (one of two agents or another one else). We then define a recruiting role which has the responsibility to maintain the knowledge of each agent upon the structure of the organisation. This means that an agent playing this role has to detect the interactions situations. The internal architecture of reactive agents match the constraint of physical phenomena disconnection presented before, because an agent can have several reactive behaviour, each one participating in a different interaction network. This elementary behaviour consist in the computation of a vector of internal state variables after the evaluation of inputs (from the interactions where the agent is a target) and presents a pertinent internal state to other agents (potentially targets of an interaction where the former agent is a source).

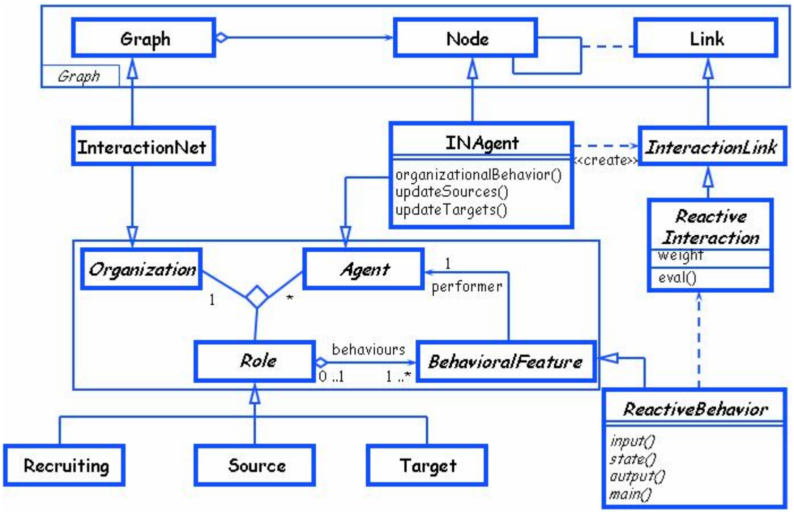

Figure 2. The Interactions Network.

The social environment.

The social environment is also populated by more "intelligent" agents. They undergo it and they act on it as reactive agents, but the way they choose their actions is carried out on a higher level of abstraction. Those agents are various humans involving in the formation (learners and teachers) who are played by autonomous agents. In our case, the social environment is structured and each member knows its roles and those of its partners. The interactions between the team members are also structured and arranged by the mean of procedure known by all members. We thus derive our generic organisational model to formalize this concept of team. We are interested in the case where the action's coordination between team members are already envisaged and written in procedure. On the other hand, the environment being dynamic, agents can need to adapt the scenario to the environment. The procedure must then have a semantic representation so that agents can reason above. Then we use a high level language to describe a procedure (Allen temporal logic).

The reasoning of team members relates on organisation, procedures and actions. We propose a model of agent having local organisational knowledge. A rational agent is divided into a decisional part and a part represented by modules of perception of the physical environment, communication and actions (Figure 3). It must carry out actions of the procedure and adapt to situations not envisaged. The procedure describes interactions between agents in an optimal case, and leaves to the agent the responsibility to build implicit plans (not clarified in the procedure) considered as natural within an applicative situation. Moreover, the procedure organize actions of a semantic level which we call « actions trades » such as « sprinkling a fire » in the case of firemen procedures, whereas the implicit plans arrange actions of a generic semantic level for humans such as « going at a point ». For that, the agent must be able to reason on actions and we propose a model of goal directed actions having pre-conditions and post-conditions. Thus, before carrying out an action, the agent must make sure that pre-conditions of this one are checked. If it is not the case, it builds itself a plan by back chaining on pre-conditions and post-conditions of actions. When an agent starts or stops actions, it broadcasts a message that enables other members to follow the evolution of the procedure. When this behaviour is at fault, the agent calls its organisational behaviour which can help it to find a solution with another team member. Thus, in a hierarchical organisation, when an agent has a problem which it cannot solve, it refers to its superior. Then, the superior has the responsibility to find a solution among his subordinates. If it does not find any, it refers to its own superior about it. We represent this mechanism by a method like a *Contract Net Protocol*.

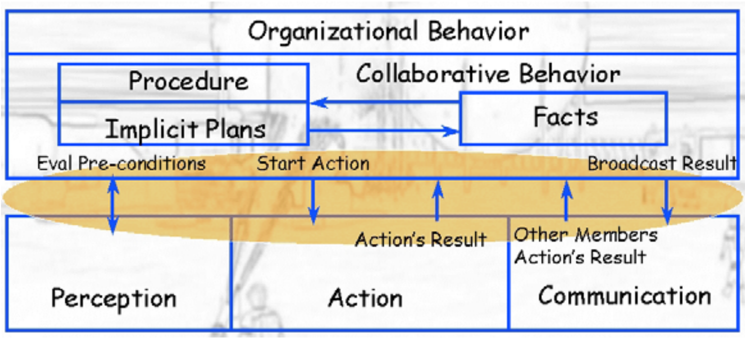

Figure 3. Architecture of rational agents.

2.3. The users

The avatar, in *MASCARET*, is not only the representation of the user but has also autonomous behaviour (reactive and rational). Therefore the avatar model in *MASCARET* is the same as rational agent. In order to provide the decision making responsibilities to the user (student) the link between the collaborative behaviour and the action module (`Start Action` message) can be inhibited. This inhibition is dynamic, means a learner can take the control of an autonomous agent during the simulation. It becomes then its avatar. At any time, the user can release its control on the avatar.

All modules composing the avatar are still active and thus remain potentially usable. The `Action's Result` message provides information to the `Facts` database. The avatar has the knowledge on the action historic and on the current action running. In addition, the avatar is still informed about result of other member's actions and still informs them about action done by the user. The knowledge on the evolution of the collaborative procedure is still consistent.

As the collaborative behaviour of the avatar is inhibited, it does not call any more the organisational behaviour (in case of a failure in an action). It becomes the responsibilities of the user.

2.4. Pedagogy

Our objective is to integrate a differentiated pedagogy based upon context. Intelligent Tutoring System (ITS) aims at providing students with dedicated tutoring. We use ITS as a system allowing to communicate a know-how. ITS can be composed of fourth components: domain model, learner model, pedagogical model and interface model. It usually uses only two models of those fourth components (domain and student) as STEVE [8]. Pedagogical and interface model is not often well specified. *MASCARET* allows the establishment of those models necessary to the creation of Intelligent Tutoring System.

The domain model.

The domain model is the representation of the teacher's expertise. It can also be called expert model. In our case, it is constituted of the set of all organization. It has knowledge on the physical environment using the concept of `InteractionNet` and the knowledge of the social environment using the concept of `Team`. The domain model needs only knowledge on organizational structures and not on the instances of those structures. For example, the domain model reference only one instance of each sort of `Team`, but never reason on the agents playing roles in this organisation.

The structure of the social environment provides procedures and actions. The expert model knows procedure layouts and goals. Concerning actions, it can consult their goals, pre-conditions and fail conditions. The social environment provides also information on roles. Therefore, the domain model is able to consult roles responsibilities. The structure on the physical environment provides links dependency (source and target) as well as reactive behaviour involved in those interactions.

The learner model.

The learner model provides, at each instant, the state of the student's knowledge. In *MASCARET*, it has three components: the `PsychologicalFeatures`, the avatar and the `Curriculum`. `PsychologicalFeatures` are information concerning student's level (novice, expert), emotion state (stress) ... The `Curriculum` associates `Exercises` and `Errors`. An `Exercise` is constituted of a `Context` and a `Scenario`. The `Context` instanciates physical and social environment. The avatar provides the collaborative procedure and student's historic and current action knowledge.

The pedagogical model.

The pedagogical model defined pedagogical strategies [15]. It contains information about how to teach knowledge. Such model has knowledge on pedagogical strategies issued from psychological and didactic research. We are particularly interested in the co-operative strategy [16], where pedagogical actors will cooperate for the realization of tasks, exchange ideas on the problem and share the same goals. Moreover, we noted the disturbing strategy [17] where the goal is to disturb the learner by proposing solutions that can sometimes be erroneous. That way, we force the learner to evaluate his self- confidence in his own solutions.

The errors model.

We consider that errors are crucial information. Therefore we decided to increase the number of models with the errors model. This model contains pedagogical information on general errors (errors classically done by student). As a tutor has the knowledge on the domain (domain model) and on the student (learner model), we are able to note an error on the procedure. We envisaged to add errors on general rules that student should respect during the entire simulation.

The interface model.

The interface model permits the communication between the system and the student. It presents information to the student. It defines where and how exposes pedagogical information. It also selects with which resource pedagogical information will be exposed. In addition, it is able to detect student's action.

The pedagogical agent.

In our case, we use the avatar to play a pedagogical role because it has the knowledge on the domain (domain model) and the student (learner model). In order to get a pedagogical behaviour, the avatar inherits from pedagogical agent. Our perspective is to integrate pedagogical agents, taking different roles. Therefore such actors have a common objective: to increase the student's skills. During the simulation, a pedagogical agent can take the decision to intervene.

Pedagogical action inherits from actions in order to keep deductive reasoning and to store dated information for the debriefing. We use generic pedagogical actions as `Explain`, `Suggest`, `Show` or `Disturb` (Figure 4). The pedagogical action `Explain` provides current action goal. Moreover, it is able to provide the sequel of goal achieved so far. The pedagogical action `Suggest` informs the user about the next action. It provides goal and pre-conditions on the next action. The pedagogical action `Show` simulates such action. Concerning the pedagogical action `Disturb`, the learner acts in the framework of a mission. The domain model informs about the goal of this mission, which contains information about an instance of `InteractionNet`. The domain model also informs about the interaction in this `InteractionNet` and the `Context` informs about agents playing roles in this organization. Disturbing the learner is achieved by modifying the behaviour of the agents playing the `Source` role in this `InteractionNet`. For example, the pedagogical action `Disturb` can consist to modify the orientation of the wind in order to show up gas propagation.

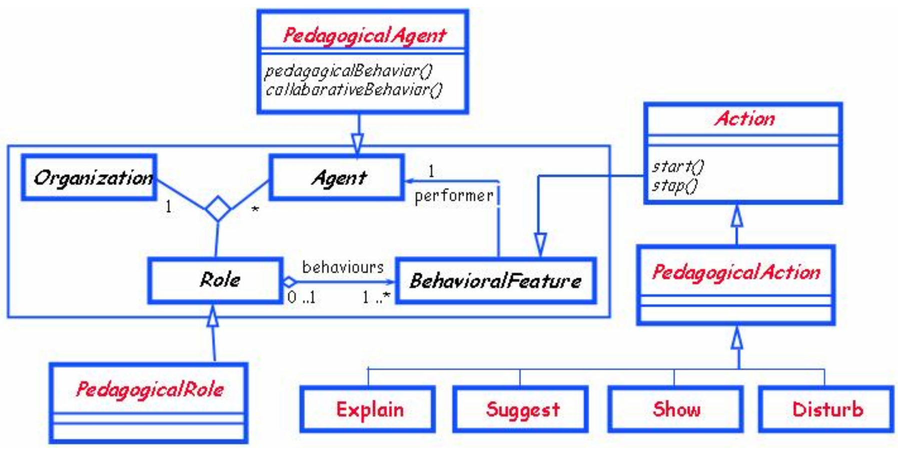

Figure 4. The pedagogical agent / role / actions.

The pedagogical behaviour.

Using the fourth models, a pedagogical agent can help student with dedicated tutoring. We proposed a behaviour using interactions between the five models (Figure 5). The pedagogical behaviour is a five steps cycle. First, the pedagogical agent observes the student's action using the interface model. The avatar of the student has the knowledge on the current action running. Such observation permits to update the learner model.

Second, we compare the expert model to the learner model. That way, we are able to detect an error. For example, if the goal of an exercise is to respect a procedure. The student achieves action A and starts action C. The expert model provides information on the procedure. Therefore we know that after action A the student should start action B. The comparison detects an error corresponding to a layout.

The error detection uses the error model and updates the learner model. In addition, we can increase the errors model. According to information on the error and using the pedagogical model (strategies), the learner model (level, emotion state ...) and the `Context` (Figure 4) the pedagogical agent selects a pedagogical action. Finally, the pedagogical action selected is represented to the student using the interface model.

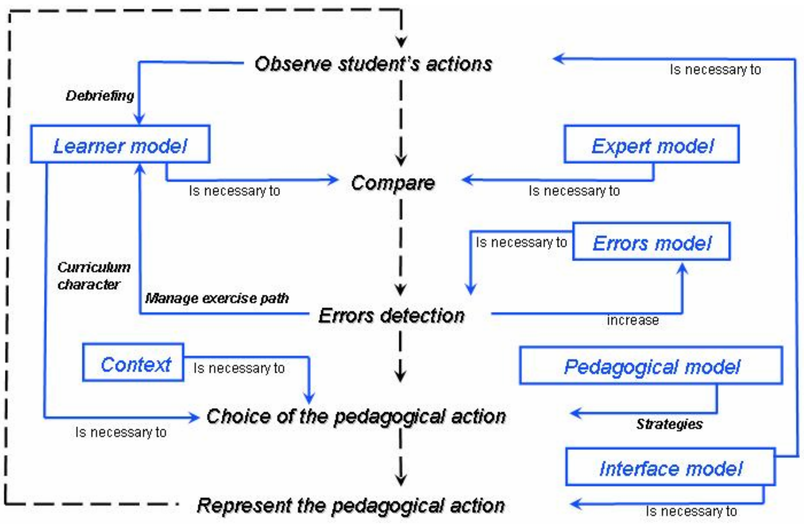

Figure 5. The pedagogical behaviour.

3. SécuRéVi

SECUREVI (Security and Virtual Reality) is an application of *MASCARET* to civil security. It helps with the training of firemen officers for operational management and for commandment. A complete description of *SECUREVI* is presented in [18].

The physical environment is constituted of the site where exercises take place as well as physical phenomena (fire, smoked, jets of water...) being able to intervene. The learners play roles of different leaders intervening during the incident. Teachers participate in the simulation to trigger malfunctions and help the learners or play a role in a team. The designer work in *SECUREVI* is essentially to implement elements of physical and social environment by inheriting *MASCARET*. Thus, the designer has to conceive his own agents to simulate a specific phenomenon. This is possible by inheriting `INAgent` as well as its reactive behaviour (inheriting `Reactive Behavior`). Then he defines the networks interactions (`InteractionNet`).

The conception of social environment composed of FTP teams (Fire engine Thunders Pumps) in charge of the incident attack or CMIC team (the Intervention Chemical Movable Cell) follows the same path by inheriting model of social environment proposed by *MASCARET*. Thus, the designer has to describe new teams and roles as well as actions that agents have to perform. *SECUREVI* is implemented using the platform *AREVI/ORIS* [19].

The domain model consists then of instances of FTP teams to get the knowledge on the social environment and instances of all physical phenomena.

The learner model has avatar (section 2.3) to get knowledge about student historic and current action running.

The avatar (see section2.3) is a pedagogical agent and plays a pedagogical role. It can `Explain` how and why something must be done during the exercise. The avatar can also `Suggest` what should be done next. The system uses speech generation to `Explain` or `Suggest`. In addition, the avatar can `Disturb`. It uses the physical environment. For example, if goal of this mission is to stop gas propagation, disturbing consists to show up gas propagation. That means the pedagogical action `Disturb` increases the wide power or changes its orientation. In fact, thanks to different level of interaction network, physical phenomena could be adapted to the student level. It is not necessary to display every disturbing element for a novice. As opposite, it is possible to add new physical disturbing phenomena for an experimented student in order to improve his skill. The physical environment also allows showing elements that are not visible in the real [20]. For example, we could display wind curve or gasses cloud in order to specify to the student specific conditions.

In future works, we will implement pedagogical agents playing pedagogical roles (Figure 4). We consider that every agent can play a pedagogical role (firemen, gas, water jet, hose reel ...).

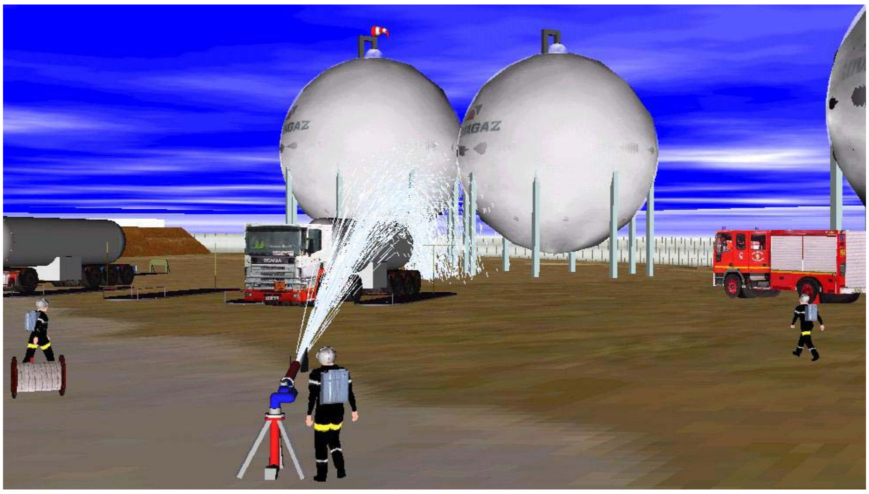

Figure 6. Picture from *SÉCURÉVI*.

4. Conclusion and future works

Our objective is to place learners in operational positions by simulating their physical and social environment. Considering a virtual environment for training as a multi-agents system, we propose the model *MASCARET*. It allows the realization of a realistic, cooperative and adaptive virtual environment for training. We have defined the context of simulation based on physical and social environment. We have presented how we integrate pedagogy in *MASCARET*. We have defined a pedagogical agent (avatar) playing pedagogical role. Finally, we have proposed a pedagogical behaviour based on models necessary to the creation of Intelligent Tutoring System.

Using the generic organisational model of *MASCARET* (Figure 1), we envisaged to define a pedagogical organization constituted of pedagogical agents playing pedagogical roles (Figure 4). We envisage several roles. First, we are interested by the tutor. Using the notion of procedure as describe in the social environment, the tutor can provide a deductive reasoning and explications [8]. Therefore a simulated tutor can show a demonstration of the next action in line. Second, we propose the use of the role of companion [17]. The companion is a virtual actor that will cooperate for the realization of tasks, exchange ideas on the problem and share the same goals. We also are interested in the role of troublemaker [16], where the goal is to disturb the learner by proposing solutions that can sometimes be erroneous.

Future works will define an accurate specification of what errors are and how to use pedagogical strategies. Pedagogical strategies guide pedagogical agent in the choice of pedagogical actions. If it is considered we are in possession of such different pedagogical strategies, our ITS will be multi strategic [15]. In order to achieve a pedagogical goal defined by a pedagogical strategy, we envisaged to integrate collaboration between pedagogical agents using pedagogical organization.

Finally, we want to provide to our system the possibility to adapt pedagogical behaviour to a specific student. In this optic, the choice of a pedagogical action will be adaptive. In our model of behaviour (Figure 5), the selection of a pedagogical action depends on the error, the context, the learner model and the pedagogical model. We envisage the use of machine learning techniques as classifiers systems [21].

5. References

[1] Piaget, J., *Le comportement, moteur de l'évolution*, Gallimard, Paris, 1976.

[2] Fuchs, P., Moreau, G. and Papin, J., *Le traité de la réalité virtuelle*, Les Presses de l'Ecole des Mines, 2001.

[3] Tisseau, J., *Virtual Reality: in virtuo autonomy,* Accreditation to direct research. University of Rennes 1, 2001.

[4] Demazeau, Y., From interactions to collective behaviour in agent-based systems, *Proceeding of the European Conference on* Cognitive Science, 1995, pp 97— 132.

[5] Clancey, W.J., Simulating activities: Relating motives, deliberation, and attentive coordination, *Cognitive Systems Research 3(3),* 2002, pp 471— 499, special issue on situated and embodied cognition.

[6] Lourdeaux, D., Fuchs, P., Burkhardt, J.M., An intelligent tutorial agent for training virtual environments, *Proceeding of 5th world multiconference on Systemics, Cybernetics and Informatics SCI'01*, 2001.

[7] Gerval, J., Popovici, M., Ramdami, M., El Kalai, O., Boskoff, V. and Tisseau, J., Virtual Environment for children, In *Proceedings of the 5th IASTED International Conference "Computers and Advanced Technology in Education" (CATE'02)*, 2002, pp 416— 420.

[8] Rickel, J. and Johnson, L., Animated agents for procedural training in virtual reality: Perception, cognition, and motor control, *Applied Artificial Intelligence*, 1999, 13.

[9] Py, D., Quelques méthodes d'intelligence artificielle pour la modélisation de l'élève, *Sciences et Techniques Educatives*, 1999, 5(2).

[10] Woolf, B.P., « Building Knowledge Based Tutors », TOMEK I., Ed., Computer Assisted Learning, *Proceedings of the 4th International Conference (ICCAL'92)*, 1992, pp 46— 60.

[11] Nkambou, R. Gauthier, G. and Frasson, C., Un modèle de représentation des connaissances relatives au contenu dans un système tutoriel intelligent, *Sciences et techniques éducatives*, 1997, 4(3), pp 299— 330.

[12] Self, J., *Artificial Intelligence tools in education*, chap. Student models: what use are they ?, 1998, pp 73— 85.

[13] Hannoun, M. Boissier, O. Sichman, J. and Sayettat, C., MOISE: Un modèle organisationnel pour la conception de systèmes multi-agents, *Actes des Journées Francophones en Intelligence Artificielle et Systèmes Multi-Agents* (JFIADSMA), 1999, pp 105—98.

[14] Gutknecht, O. and Ferber, J., Vers une méthodologie organisationnelle de conception de systèmes multi-agents, *Actes des Journées Francophones en Intelligence Artificielle et Systèmes Multi-Agents* (JFIADSMA), 1999, pp 93—104.

[15] Mengelle, T. and Frasson, C., A multi-Agent Architecture for an ITS with multiple strategies, *Actes de la conference internationale CALICE'96*, LNCS 908. , 1996, pp 96—104

[16] Aïmeur, E., Frasson, C. and Dufort, H., *Cooperative learning strategies for intelligent tutoring systems*, Applied Artificial Intelligence 14., 2000, pp 465 — 489. .

[17] Chan, T.W. and Baskin, A.B., Learning Compagnion Systems. In C. Frasson & Gauthier (Eds.), Intelligent Tutoring Systems: *At the Crossroad of Artificial Intelligence and Education*, 2000, Chapter 1., pp 159 — 167.

[18] Querrec, R., Buche, C., Maffre, E., and Chevaillier, P., SécuRéVi : virtual environments for fire-fighting training. *Proceedings of VRIC 03*, 2003, pp 169 —175.

[19] Harrouet, F. (2000). *oRis : s'immerger par le langage pour le prototypage d'univers virtuels à base d'entités autonomes,* PhD thesis, Université de Bretagne Occidentale.

[20] Mellet d'Huart, D., Richard, P. and Follut, D., Virtual reality for education and training: State of science and typology of uses, In Richir, S., Richard, P., et Taravel, B. editors, *Proceeding of VRIC401*., 2001, pp 47— 55.

[21] Wilson, S.-W., ZCS: A zeroth level classifier system, *Evolutionary Computation*, 1997, 2(1), pp 1— 18.