Contemporary Engineering Sciences, Vol. 3, 2010, no. 2, 63 - 76

Integration of Pedagogical Agent and Simulator of Interactive Pedagogical Activities: SIMUL*Agent*

Zineddine LAOUABDIA-SELLAMI

Department of Computer Science
University of Annaba
P. Box : 12 Annaba 23000, Algeria
zd-laouabdia@yahoo.fr

Tahar BOUHADADA

Research Group on Artificial Intelligence(GRIA/LRI)
Department of Computer Science
University of Annaba
P. Box : 12 Annaba 23000, Algeria
bouhadada.tahar@univ-annaba.org

Abstract

The Research in Human Interactive Learning Environment (HILE) evolves in three directions: adaptive hypermedia which allow a presentation of educational hypermedia documents adapted to the learner's level, pedagogical simulator in a distance learning context and animated pedagogical agents which aims at explaining the course and motivating the learner by communicative and emotional aspects through verbal and non-verbal communication signs.

In this paper, we present a new approach aiming to integrate a pedagogical agents and pedagogical simulator (cooperation in distance learning context) which are in fact complementary but not yet satisfactorily integrated in the current systems.

Our global gait is based on studies emanating from interpersonal multimodal communication domain, and on analyses of the multimodal behavior of the teachers facing the learners. In this paper we describe an environment permitting, for a given pedagogical situation (learner's level, content to present), to distribute the adaptative presentation between the simulator and the pedagogical agent.

Keywords: Interactive learning activity, pedagogical agent, pedagogical simulator, multi-modality, cooperation

1 Introduction

In traditional learning environment, the interactions between learner and teacher (or between learners working in group) are multi-support (drawings in blackboard, books, notes, audiovisual documents), multi-modal (words, gestures, postures, glances), cooperative (exchanges having a common goal) and adaptive (the teacher can modify the options of its course to reply to the questions). One of the teachers' difficulties is to adapt his communication to learners which have different levels. With consideration of this multimodality and the cooperative and adaptive nature of these human interactions, the current tools of the Interactive Environments for Human Learning (IEHL) seem limited [21].

Several works have studied the impact of the multimodality, the adaptability and the cooperativeness on education (quality of learning, learner's attention, interactivity of the course, adaptation to various learner's level and paths (MICAME project) [21] which aims at understanding the nature and the mechanisms which underlie these multimodal and cooperative interactions in order to consider the specification of multi-modal Human-Machine Interfaces (clarification, empathy and encouragement), more convivial, more pedagogical, more effective but also allowing to adapt not only the contents of the course for various learners level, but especially the communication of this course [1][21].

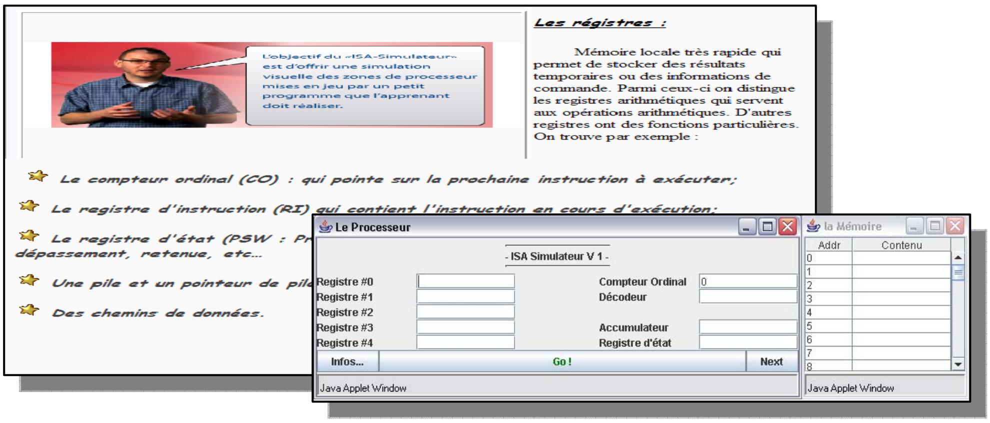

Figure 1. The two main components of the system

In this paper, we describe a system integrating a simulator of interactive learning activities [3[13] and a conversational agent [2][11] (Figure 1). We focus on problems arising from the cooperation to be implemented between these components which had been until now developed independently without integration in an application. Each of these two components must integrate the adaptivity in relation to the learner's knowledge level and the historic of his/her behavior and consultations of the course. Then, the problem consists of coordinating these two adaptive components in a transparent and intuitive way for the learner.

2 Related works

Education is considered as a privileged domain for the application of the simulators of interactive learning activities [5]. The objective in this context is to present multimedia pedagogical simulations in an individual way, adapted to the learner. This adaptation is generally carried out according to an acquired knowledge which is determined by the system by validation of steps, tests and/or according to the list of the pedagogical elements already consulted. The general architecture of the learning systems is based on three components: a) a learner's model which represents the information known by the system about the user. This information can be introduced directly by the user, determined by results of tests or exercises, or have been deduced from the user's behavior during the interactions (choice of navigation, time of consultation, etc.) b) a domain model that structures the pedagogical elements that the user can access. The domain is generally structured according to a hierarchy of concepts which represent pedagogical entities that the learner must validate. These concepts are provided with a set of relations (for example: pre-requisite); c) an engine which determines which pedagogical elements and which possibilities of activities are proposed to the user at a given time.

These three components are closely dependent; the domain model is structured according to the type of processed information by the learner's model [21]. Some works use XML to represent the domain model and/or the learner's model as well as to describe the structure of the pedagogical documents [4].

The pedagogical agent is situated at the crossing of two directions of research: conversational agents and knowledge based learning environments. The pedagogical agent is an animated character, posted beside a formation support or integrated in a 3D environment, which aims to provide some nonverbal signs of communication (look, facial expressions, gestures of the hand, stances) while trying to motivate the learners by communicational and emotional aspects [15]. Several teams are currently developing pedagogical agents which differ according to their graphic features (just a talking head or a complete body, 2D or 3D) and the algorithms that they use to generate a multi-modal behavior according to a communicative task.

The multimodal implementation of the pedagogical acts can be attenuated or can be exagerated according to the context. The non-verbal modes are used by the conversational agents to control the turns of speech and to provide a return without interrupting the user, to punctuate a sentence, to put an emphasis, or to indicate a question [8][19]. In addition to these general communication acts, the pedagogical agents can use a non-verbal communication to show how to execute a physical action in a 3D simulated environment (gestures of the hand, stances), to carry the learner's attention to a certain particular aspect of a complex environment, source of ambiguousness, to congratulate him (facial expressions, gestures of the hand), to encourage him to ask questions (gesticulating, scraping the head) [22][16].

Several studies have been carried on to evaluate the utility of these pedagogical agents on the apprenticeship [17][9]. Some have observed the agent's impact for example on the student’s motivation and on the quality of the learning transfer [18].

These different projects have some limits. Thus, the pedagogical agent’s behavior and application are generally integrated in the software and do not benefit from the more recent progress in term of specifications (for example by using XML) more easily re-usable to generate different courses. On the other hand, the possible use of agent to personalize, in a generic way and independent of the pedagogical application, the presentation of the course for the specific needs of a given learner (for example with different personalities of agent) or through adaptive capacities of the pedagogical application is little described. Finally, the multimodal behavior of the agents is often founded on general rules drawn from social-linguistic literature and not on a contextual analysis of the multimodal communicative strategies of teachers observed in learning situation [21].

3 Simulation in learning process

Simulation is a modeling technique of the reality. It allows to represent the functioning of a system involved of different centers of activities, to put in evidence their features and the interactions between them, to describe the circulation of the different messages and objects treated by the process, and finally to observe the behavior of the system as a whole and its evolution in the time [10]. In the pedagogical context, the word ‘simulation’ includes all pedagogical situations where the learners must hold a role and act consequently, corresponding to a scenario extracted from the reality [7].

3.1 Pedagogical simulation

The free use of simulation by the learner doesn’t guarantee the learning. It is necessary to give the learner concrete goals, and the simulation must permit to guide the learner toward the fixed goal [7]. In a traditional learning situation, the teacher can fix the objectives and directs the learner during his manipulation of the simulation [24][14]. On the other hand, in autonomous use of simulation, the system should be able to propose the objectives, to guide the learner toward the objectives and to verify if the objectives have been achieved. It becomes then necessary to be able to control the learner's activity pedagogically during his manipulations of the simulation and to provide a set of tools dedicated to this control.

In this work, we will only be interested in the type of simulation that is located in the category "to simulate to learn", while excluding all other categories of simulation where pedagogical check of the learner doesn't exist.

For all pedagogical simulation, it is necessary to specify some concepts [20]:

- What are the features of the knowledge that the learner must acquire because of the simulation?
- In which pedagogical contexts do the simulations can be used?
- What is the goal pursued by the learner when he uses a pedagogical simulation?
- What type of pedagogical control does the computer permit to achieve?

3.2 Learning Active Situation (LAS)

Pedagogical simulations are softwares which propose Learning Active Situations (LAS) based on simulation. In the present work, we are interested to implement learning active situations (activities controlled by the student) in a distance-learning context, exploiting a simulator of interactive pedagogical activities. It consists at giving teachers tools to propose and to manage the learner's activity, and to provide too, the different actors, with means of communication and exchange.

3.2.1 Conception of Learning Active Situation

For developing the Learning Active Situations, we adopted the M.A.R.S model (Model, Association, Script, Representation) [20]. This model is a general conceptual setting that guides the designer in his process for defining the pedagogical simulation application. Its objective is to make palpable the notion of distinct workspaces in which intervenes every actor, provided of well listed expertise. Its goal is, therefore, to serve as a reference of organization for the application software development [13].

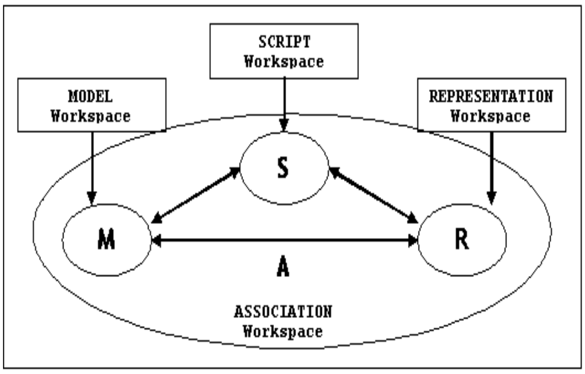

Figure 2. Workspaces in M.A.R.S model

M.A.R.S model decomposes a pedagogical simulation to four workspaces (Figure 2):
- a *Model Space (M)*: describes the simulated system;
- a *Representation Space (R)*: defines the interfaces manipulated by the user;
- a *Association Space (A)*: permits the integration of the Model and the Representation in order to get a simulation;
- and a *Pedagogical script Space (S):* a Script directs the learner control during the session, so we call it controlled simulation.

4 The pedagogical agent

It is represented by an animated character and located beside a formation support, which aims to provide communication signs for assisting and to motivate the learner by communicational and emotional aspects.

It can:
- attract the learner's attention,
- motivate the learner (by encouraging him),
- provide assistance to answer better,
- give explanations and indications,…
- etc...

In SIMUL*Agent*, the pedagogical agent intervenes in the different phases of the learning process:

- In "Course" phase:
- to provide detailed explanations,
- and can also intervene, if it is requested by the learner;
- etc…

- In "Exercises" phase:
- to provide assistance,
- to guide the learner,
- etc…

- In "Simulation" phase:
- to explain the steps to follow for simulating the given program,
- to contribute to the simulation;

5 Description of SIMUL*Agent* system

The system consists of a domain model, a learner's model and a controller which determines the course's elements to be presented, on one hand by the simulator and, on the other hand by the pedagogical agent (Figure 3). This separation aims to permit a re-use for different courses and supports.

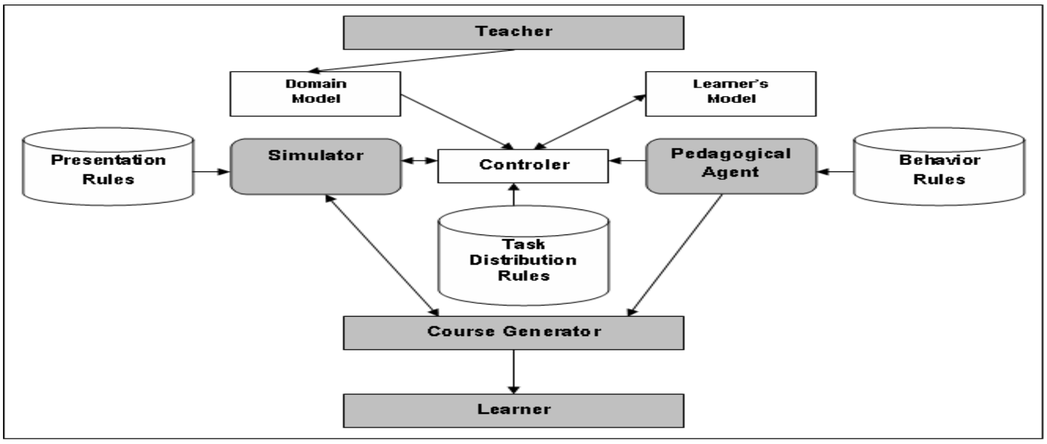

Figure 3. The architecture of SIMUL*Agent* system

The objective of our work is to study the integration of a simulator of interactive pedagogical activities and a pedagogical agent for an adaptive learning system. In a first time, we consider a simplified learner's model which takes into account an evaluation of the learner's level (for example seized initially or resulting from preliminary evaluations): *Weak, Average, Good*; a set of concepts with Boolean values (not validated, validated), and the historic of the steps covered in the pedagogical simulation. The pedagogical documents presented to the learner must be interpreted on one hand by the simulator and on the other hand by the pedagogical agent. Therefore, their representation in the system must not be limited to the level of the details presentation in one or the other of the exit media, but to the level of the documents structure. We opted for an XML representation: the course is organized in pedagogical documents called concept, every concept is can contain the following elements: definition, explanation, and example. For each course element, can be associated a set of pre-requisite concepts, and a lowest level required.

Modalities	Only the simulator
	Agent and Simulator
Level of details of the communicated information	Simulator: Simple/more detailed
	Agent : Simple/more detailed (number and length of the interventions, non-verbal activity)
Step validation	Very directed by the system
	More liberty of step validation let to the learner

Table 1. Measurements on which the cooperation of the system can intervene

We chose three measurements on which the cooperation of the system can intervene according to the learner's level and his consultation frequency of the course (Table 1): the used modalities, the detail of the communicated information and the validation of the steps.

These measurements regroup on one hand, those classically studied in the multimodal interfaces and, on the other hand in simulators. Cooperation is managed at different level: in the selection and in the organization of the communicative steps, but also in the internal implementation of each of these steps. At the first execution of the system, the algorithm presents the following communicative steps: welcome message, introduction to the course, selection of the 1st concept (by the system or by the learner). Then, for each selected concept, the steps are the following: *introduction to this concept, presentation of the content of the concept, presentation of the steps of the simulation.* As an example we explain the functioning of the system during the presentation of the course section. A concept is composed of elementary units.

For all concept unit *unit*:

If *unit*.level < = learner.level **then** *unit* will be presented in the simulator
If *unit*.level = = learner.level **then** *unit* will be presented via the pedagogical agent
If *unit*.level > learner.level **then** *unit* will not be presented.

6 ISA-Simulator

"ISA-Simulator" is a simulator integrated in the SIMUL*Agent* environment, which can simulate the microprocessor functioning. The objective is to offer a

visual simulation of the processor zones.
"ISA-Simulator" consists of a reduced set of elements that we usually finds in a microprocessor (registers, accumulator, instruction counter, status register,…etc)(Figure 4).

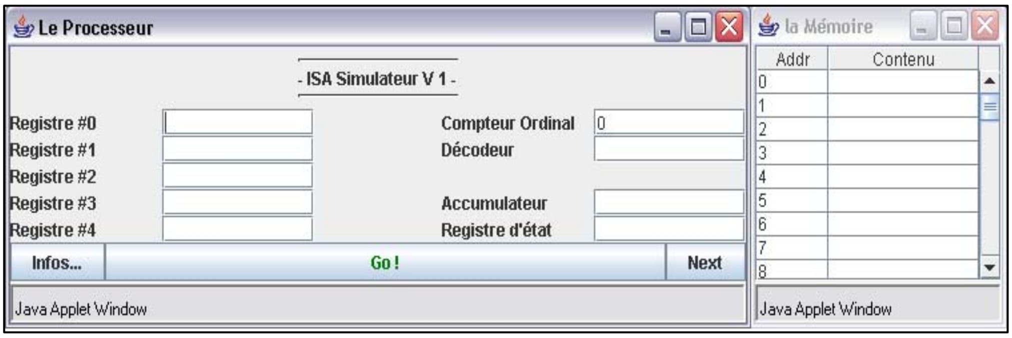

Figure 4. ISA-Simulator interface

ISA-Simulator includes:
- 5 registers numbered from #0 to #4,
- an accumulator to recover the computation result,
- an instruction counter which point on the memory address which contains the instruction to execute,
- a decoder that displays the instruction in progress,
- and a status register which alerts on the important events (case of a negative result (N), or nil (Z)).

It offers a small memory of 50 boxes, in which we can enter data and instructions, the box memory address is noted with $.

The learner has the description of each instruction, and must complete some exercises while conceiving a program. Then he/she introduces the program and can launch its execution. The simulator obeys to the instructions, and puts in evidence the zones which work in the microprocessor, step by step, visually (Figure 4).

7 The pedagogical script

A script explicitly defines the different activities proposed to the learners on the simulator. It also specifies the control that will be made on the learner's progression during this activity; it determines the pedagogical assistance that will be provided to him/her automatically according to his/her progression [13].
For this effect, the simulator's interface is provided with a panel including different descriptions of every step of the work to be treated.
A pedagogical script is defined by [13]:

- the initial situation and the objective to reach;
- situations corresponding to pertinent resolution steps;
- particular situations to observe (constraints to respect, classical errors, potential dangers,...);

- the reactivity that enables the assistance to the learner according to his/her progression. It determines the system reactions (feedback information, assistance, return back to the first step,...) corresponding to different controls (successful step or not, particular reached situation, objective reached or not).

When it is executed, such a script does not force the learners to reach the objective according to a unique progression, but enables the validation of a set of solutions. Indeed, the teacher fixes the steps that he/she judges inescapable as well as their order; the system will verify if the situations corresponding to these steps will be achieved successively, but will not control the way that the learners proceed to reach the end of every step. However, the system will be able to detect the particular situation reached by the learner if the teacher specified it as a situation to be observed.

Once defined by the teacher, the script enables the proposition of different work modes to the learner: "*learning*" mode or "*evaluation*" mode [25]. In the present work, we exploited the *learning* mode. In this mode, besides the global introduction, the learner receives the instruction of the 1st step and can have access to an assistance provided for this eventually. When he/she thinks that he/she succeeded in this step, the learner asks for the validation of his task and receives the corresponding evaluation. As soon as an intermediate step is gone through with success, the order of the next step is displayed. The learner can abandon a step at any time and can go to the next (the system puts the simulation as if he/she had succeeded the step). If he/she reaches a situation "*to be observed*", the feedback information and the expected action are then launched.

7.1 The pedagogical script model

The script includes various descriptions of steps described in an XML format. It is defined by its title, a general statement, and the different resolving steps (one step at least). The step is considered like a sub-objective to reach and the learner must successively validate all the steps to succeed the exercise. However, he/she can ask to abandon the current step and go to the next. Feedback information is provided in case of global *Success* of the exercise as well as in case of *Failure* (Figure 5) [12].

Every step is characterized by:

- *A description*: this indicates briefly the essential objective of the step. It enables to mark steps;
- *A statement*: it contains the specific instruction of the step;
- *An initial state*: it is used by the script monitor to initialize the simulator in the beginning of the session.

- *Features of the step validation*: they contain a situation that enables to validate the step as well as the associated reactivity in case of *Success* as in case of *Failure*. When the learner asks to validate the step, the script monitor evaluates if the current state of the simulator satisfied these conditions and thus determines if the step is gone through with success or not.

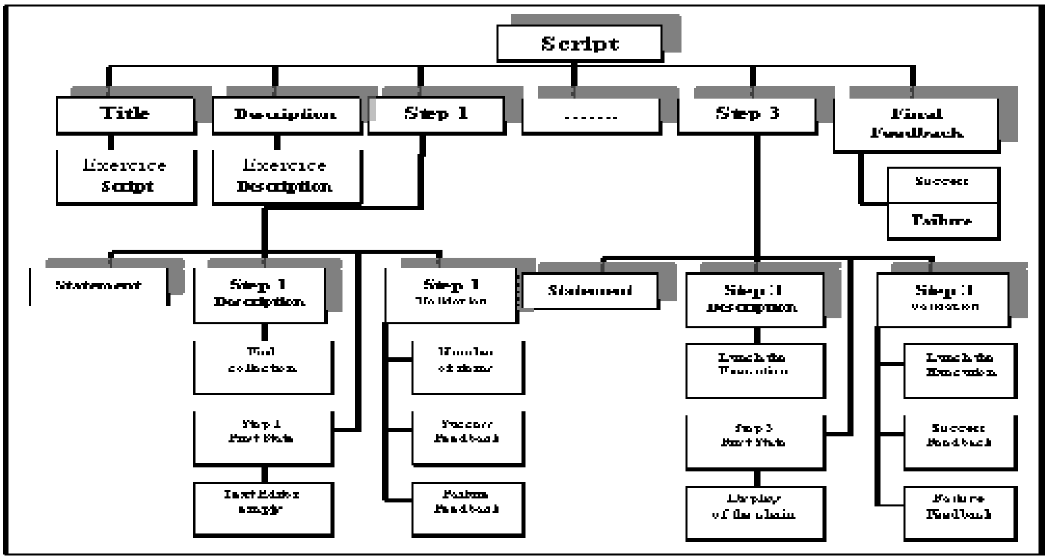

Figure 5. Arborescent representation of a script

8 The controller

It’s the core of SIMUL*Agent* environment. Its role is to check the information introduced by the user (Learner, Teacher or Administrator).

- For the Learner

- Course Phase

- To recover the more adapted course to the learner’s level,
- To distribute the knowledge between the pedagogical agent and the simulator,
- To select and to load the acts of the pedagogical agent,
- To transmit the structure of the domain and the pedagogical act to the course generator.

- Test / Exercises Phase

- To select the list of the exercises,
- To recover the learner's answers,
- To compare the answers with the teacher's correct version and to assign the score,
- To allow the learner the access to the higher concepts (if the obtained score is higher than the average) and to update the profile (in the contrary case, it advises the learner to review the concepts).

- For the Teacher

- Update of the different knowledge bases;
- The learner's follow-up,
- …

- For the Administrator

- Management of the system,
- Follow-up of the different users,
- …

9 The course generator

The course generator is in charge of building dynamically the course's pages. The generation starts when the learner decides to activate a course.

For each link selected by the learner: in a first time, it consults the table of the courses recently consulted by the learner, if the course in question is there with an access date equal or ulterior to the date of the last profile update, he/she is allowed access directly with the same construction which is in this table and that corresponds to his last access.

Otherwise, an analysis is set, and in this situation, we can meet three cases:

1. *The knowledge level acquired in the pre-requisite is not sufficient* to allow the learner the access to this course, in this case the list of the pre-requisites is displayed to him with a link allowing him the access to the course entirely and the pedagogical agent is activated.
2. *The knowledge level acquired in the pre-requisite is sufficient* but the date of the last access exceeds the length taken into account in the oblivion factor (ten days in our case). In this case a summary of the pre-requisite in question is displayed to him with a link allowing him the access to the course entirely, followed by the course which has been analyzed.
3. *The knowledge level acquired in the pre-requisite is sufficient* and the access date doesn't exceed the length taken into account in the oblivion factor; the learner can have access directly to the course.

The list of the exercises displayed in the exercise menu corresponds to those whose courses have already been consulted.

10 Implementation

The realization of an interactive distance learning environment requires languages dedicated to the implementation of web applications. The implementation of SIMUL*Agent* required Customer oriented language and Server oriented language. The system has been developed on the APACHE base server and uses its PHP interpreter for interpreting the different interactions. For the implementation of the course's base, we used XML language, more adapted for developing hypertext systems. For the management of the different bases, we opted for MySQL whose performances are particularly indicated for this type of application. The simulator is written with JAVA. Finally, we used FLASH MX for the implementation of the pedagogical agent and its animation.

11 Conclusion

We have described an approach of cooperation between a pedagogical agent and a simulator of interactive pedagogical activities. The system we propose is based on a domain model specified in XML in which the documents are represented according to a pedagogical structure model.

A control module in charge of distributing the presentation tasks to the learner between the simulator and the agent. This work opens many perspectives for research.

The simplified pedagogical structure of the document that we took into account for the moment should be completed. The question of the influence degree of the cooperation/redundancy between the agent and the simulator in the information presented to the learner remained open. Some information can be presented by the simulator and again by the agent, other information can only be presented by one of both.

We plan to use the current prototype to achieve a evaluation study of different scenarios of distribution tasks for which we will consider a population of learners with a protocol inspired for example from [6].

In traditional learning situation, the interactions between teacher and learner make to intervene several communicative modalities (speech, gestures, glance, facial expressions, drawing on a blackboard, transparencies,…). Although the use of pedagogical videos develops in the HILE domain, they are rarely used as resources permitting to annotate and to better understand the multi-modality of the communication in this pedagogical context.

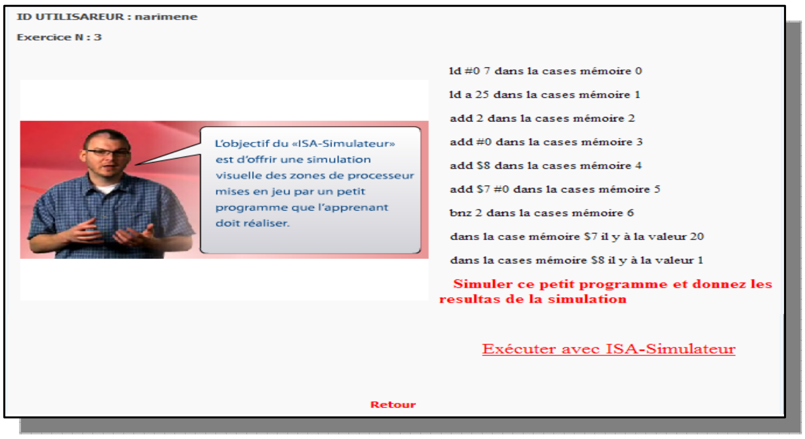

Figure 8. The pedagogical agent of SIMUL*Agent*

As we saw higher, even in the pedagogical agents domain, the multi-modal agent's behavior is often limited in expressivity and is not founded on a fine analysis of teacher's multi-modal behaviors, but rather on general rules resulting for example from the social-linguistic literature. A future work will consist in studying the contribution that the usage of video corpus can bring to determine the pedagogical scenario [23], and generally the consequences for the production of interactive pedagogical contents.

References

[1] P. C. Abrami, L'apprentissage coopératif. Théories, méthodes, activités, Chenelière Editor, Montréal, 1996.

[2] S. Abrilian, S. Busine, C. Rendu and J. C. Martin J. C, Specifying Cooperation between Modalities in Lifelike Animated Agents, In proceeding of the International Workshop on Lifelike Animated Agents: Tools, Functions, and Applications, 7th Pacific Rim International Conference on Artificial Intelligence (PRICAI'02), Tokyo, 2002.

[3] R. Azzi, Environnement de développement de simulateurs pédagogiques de procédés industriels, PhD Thesis, INSA of Lyon, France, 1995.

[4] M. E. Bonfigli, G. Casadei and P. Salomoni, Adaptive Intelligent Hypermedia using XML, in Proceeding of ACM Symposium on Applied Computing, SAC 2000, Como, Italy, 2000.

[5] P. Brusilovsky P, Adaptive Hypermedia, User Modeling and User-Adapted Interaction, (Vol. 11), 2001, 87-110.

[6] S. Buisine, S. Abrilian, C. Rendu and J. C. Martin, Towards Experimental Specification and Evaluation of Lifelike Multimodal Behavior, in Proceeding of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems, Workshop on Embodied conversational agents-let's specify and evaluate them!, Bologna, Italy, 2002.

[7] G. C. Buitrago, Simulations et Contrôle Pédagogique, PhD Thesis, University of Joseph Fourier, Grenoble 1, France, 1999.

[8] Cassell J., Sullivan J., Prevot S., and Churchill E. Embodied Conversational Agents, MIT Press, 2000.

[9] D. Dehn D and S. Mulken S, The impact of animated interface agents: a review of empirical research, International Journal of Human-Computer Studies, (Vol. 52), 2000, 1-22.

[10] F. Delepine F, Un simulateur pour répondre à des besoins de formation sur la taille de la vigne, STICEF journal, (Vol. 02), 2004.

[11] A. Galison and J. S. Nouveau , OASIF : un outil collaboratif d'aide à la scénarisation de modules de formation ouverte et à distance, Télécom Paris, Department of Educational Innovation, France, 2003.

[12] P. Gil, E-formation, NTIC et re-engineering de la formation professionnelle, Dunod Editor, 2000.

[13] V. Guéraud, J. M. Adam, J. P. Pernin, G. Calvary and J. P. David. L'exploitation d'Objets Pédagogiques Interactifs à distance : le projet FORMID, STICEF journal, (Vol 11), 2004.

[14] V. Guéraud, J. P. Pernin, J. M. Cagnat and G. Cortès, Environnement d'apprentissage basé sur la simulation, Propositions d'outils auteur et expérimentations, Research report, ARCADE project, CLIPS, France, 1999.

[15] W. L. Johnson, J. W. Rickel and J. C. Lester, Animated Pedagogical Agents: Face-to-Face Interaction in Interactive Learning Environments, International Journal of Artificial Intelligence in Education, (Vol. 11), 2000, 47-78.

[16] J. Lester, S. Converse, S. Kahler, T. Barlow, B. Stone and R. Bhogal, The Persona Effect: Affective Impact of Animated Pedagogical Agents, In Proceeding of CHI '97, Atlanta, 1997.

[17] Lester J.C., Town S.G., Callaway C.B., Voerman J.L. and Fitzgerald P. Deictic and emotive communication in animated pedagogical agents: Embodied Conversational Agents, Cassell, J., Sullivan, J., Prevost, S., Churchill, E., Eds., The MIT Press, 2000, 123-154.

[18] R. Moreno, R. Mayer, H. A. Spires and J. C. Lester, The case for social agency in computer-based teaching: do students learn more deeply when they interact with animated pedagogical agents?, Cognition and Instruction, (Vol. 19), 2001, pp. 177-213.

[19] C. Pelachaud, V. Carofiglio, B. De Carolis, F. De Rosis and I. Poggi, Embodied Contextual Agent in Information Delivering Application, In Proceeding of the 1st International Joint Conference on Autonomous Agent and Multiagent Systems (AAMAS'02), Bologna, Italy, ACM Press, 2002.

[20] J. P. Pernin, M.A.R.S: Un modèle opérationnel de conception de simulations pédagogiques, PhD Thesis, University of Joseph Fourier, Grenoble 1, France, 1996.

[21] J. H. Rety, J. C. Martin, C. Pelachaud and N. Bensimon, Coopération entre un hypermédia adaptatif éducatif et un agent pédagogique, http://www.iut.univ-paris8.fr/~martin /recherche /micame/,1998.

[22] J. Rickel and W. L. Johnson, Animated agents for procedural training in virtual reality: Perception, cognition, and motor control, Applied Artificial Intelligence, (Vol 13), 1999, 343-382.

[23] L. Verclyte, P. Useille and M. Labour, Scénario pédagogique interactif : convergence et divergence d'une écriture, In Proceeding of H2PTM'01, France, 2001.

[24] M. Vivet, Expertise pédagogique et usage de tuteurs intelligents, In Proceeding of the 13rd French-Speaking countries Workshop on Computer Science, Computer-Assisted Learning, January, 1991.

[25] E. Wenger, Artificial Intelligence and Tutoring Systems, Morgan Kaufmann, Los Altos, 1987.

Received: October, 2009