A Software System for Online Learning Applied in the Field of Computer Science

Gabriela Moise

Abstract: The computer-assisted learning is a very modern study area, which can be applied to the learning process. The main objective of this paper is to present a software system for online learning based on the intelligent software agents technologies. The main ideas on which this paper is built are: to any person is associated a learning profile (the idea is based on the existence of multiple intelligences, defined by Gardner [3]); the pedagogical resources can be shaped through educational semantic networks or through conceptual maps; a flexible software system in computer assisted learning must be based on the intelligent agents' technology. The system dedicated to computer-assisted learning must be adapted to the learning profile of each student.

The author presents a flexible online teaching software system, which learns to teach according to the learning profile of each student (the author defines this system in the PhD thesis and includes: intelligent agent structures, reward learning algorithms, algorithms to generate plans for an agent).

The application includes two agents: the supervising agent and the pedagogical agent, which determines the optimal pedagogical resources for teaching the course. The application has been designed in Microsoft Visual Studio 6.0 and uses Microsoft Agent Technology, which allows vocal recognition. Also, the Protéjé 3.0 software has been used, software that allows building ontology for computer assisted learning. The system has been experimented on the Graph Theory Course, taught at postuniversitary computer science courses, the results proving the necessity of defining a strategy for selecting the pedagogical resources presented to the students according to their learning profile.

Keywords: intelligent agent, conceptual map, learning style

1 Introduction

A software system for online learning in the field of the computer science is proposed in this paper, a system called iLearning. The system is based on shaping the pedagogical resources through conceptual maps or semantic networks and on using intelligent agent technology.[6] The main feature of the system is its flexibility.[7] In order to use this system we have to consider the learning styles of each person.[4] Donald Clark says that teaching in unconformity with the learning style of the students doesn't mean that the students don't learn, but students learn better if the teacher presents pedagogical resources according to the learning style of the students. [13] Teachers have to follow six phases in order to use the system [7]:

1. establishing the learning style of the students through an interrogation of them;

2. drawing the conceptual map of the courses using a software to build conceptual maps;

3. teaching is done based on a technique as to traversing the conceptual map of the course;

4. the teacher has to build for each node of the conceptual map a set or pedagogical resources and the system will select the best resources conforming to the learning style of students;

5. the students are evaluated in each node of the map and the teacher inputs the note in the software system;

6. the teaching takes place traversing all conceptual nodes and showing the resource to students.

The system was proposed and tested by the author of this paper in the doctoral these with the same title[7].

2   The Architecture of iLearning System

The system is composed by six functional components: Instructional Manage System, Communication System, Informational System, Evaluation System, Tutoring system, Pedagogical System. [7] The
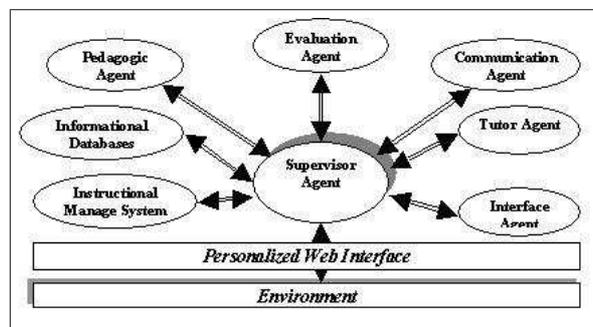

Figure 1: Architecture of iLearning System

Instructional Manage System is implemented by a SQL server and manages databases about: faculties, students, marks, credits, entrances, registration, teachers, fees; builds reports and statistics about the educational process; plans the instructional process: study periods, evaluation periods; records educational plans; starts teaching process; takes care about securing.

The Communication Agent is implemented by the software agent and accomplishes the communication between students, teachers, students and teachers and administration staff by e-mail, forum, chat; accomplishes the extern communication: e-mail, browser, chat, messaging tools.

The Informational System is implemented by the SQL server and manages the courses and pedagogical resources: records the pedagogical materials on the servers, offers tools to search courses and information in the course's databases; offers web searching tools: search engine.

The Evaluation Agent is implemented by the software agent and builds personalized tests; evaluates students according to objectives of the instructional process; evaluates the educational process: reports and statistics.

The Tutor Agent is implemented by the software agent and has the following roles: guides the teacher to build pedagogical resources; guides the students in the instruction process; guides the teacher in the teaching activities.

The Pedagogic Agent is implemented by the software agent and verifies the instructional objectives (according to Bloom's taxonomy); checks the learning style of the student; defines pedagogical strategies; defines the curriculum; checks the pedagogical resources; checks the correctness of the evaluation's activities.

The Interface is implemented by the software agent and customizes the interface. The space developed by the iLearning system is a flexible instructional environment. The work scenario of the participants to the educational process is presented in figure 2.
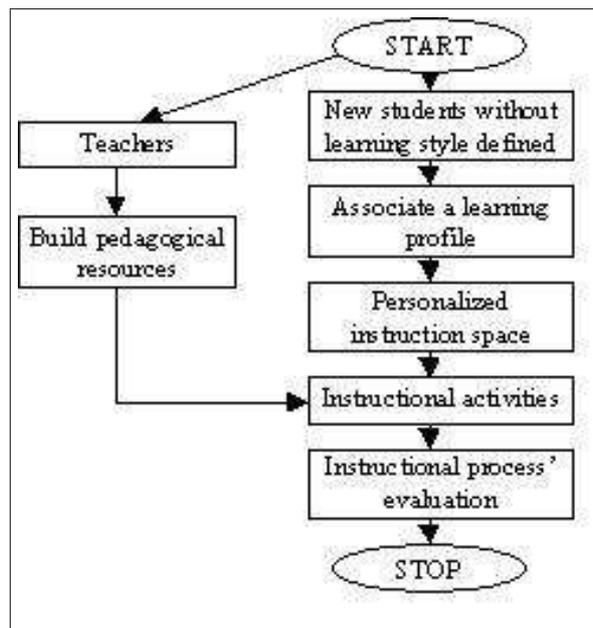
Figure 2: iLearning Work's scenario

3   The Agents of iLearning System

The iLearning system contains three kinds of components [7]:

1. "execution" agents which perform teaching-learning process. These kinds of agents don't have perception sensors; they receive the inputs from the supervisor agent. These kinds of agents are intelligent, reactive and goal oriented.

2. "supervisor agent" is a software module which has some intelligent grade. This kind of agent manages the whole system and it is able to learn how a system can be managed.

3. databases system. This component was implemented through a SQL server and VB client.

3.1   The Execution Agents

The execution agents are adapted according to INTERRAP [8], model to which was added a capacity component and a decisional module, which establishes the ability of the agent to solve a situation.

The architecture of this kind of agent is presented in the figure 3.

The control unit receives messages about the state of the environment from the supervisor agent. The unit establishes the ability of the agent to solve the problem. If the agent is not able to solve the situation then it returns an error message. The capacity is defined by the role of the agent in the system. The module analyzes the information received from the supervisor agent; recognizes the situation (goal oriented job or reactive job) and decides the level that will be activated. Finally, the module receives the actions that have to be performed by the actors or returns a message to the supervisor agent. The activities are implemented through three levels: communication level, goal oriented level and reactive level. The control unit has the duty to realize the maintenance of the plans library, the behaviors library and the knowledge base of the system.

3.2   The Supervisor Agents

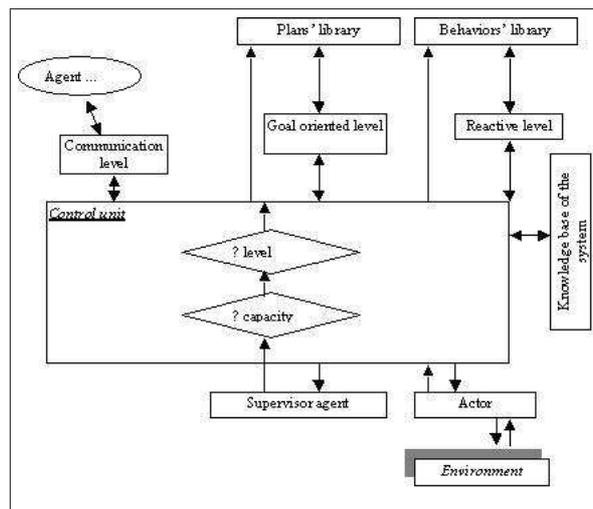The supervisor agent performs the following functions:

Figure 3: The Architecture of Execution Agent

1. coordinates the whole activity of the system;

2. receives the information from the environment (students, teachers and staff) and decides the agent that will be called;

3. receives the requests from the agents and transmits them to the destination agents or solves them using its owner resources;

4. records the capacities of the agents;

5. manages the agents' library;

6. manages the databases systems;

7. communicates with the iLearning developments.

The supervisor agent has the properties of the facilitator agents, mediator agents and broker agents. The architecture of the agent is shown in the figure no. 4.

The structure of the agent is simpler then the one of the execution agent while it performs only two types of situations: social situations and administrative situations.
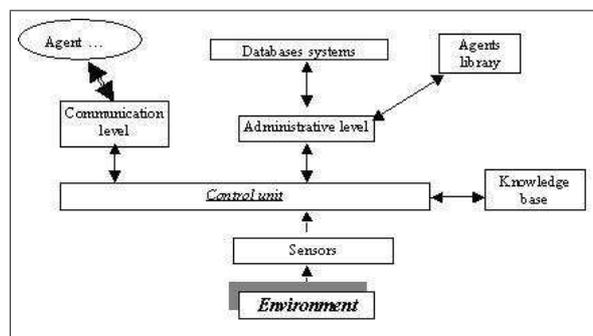

Figure 4: The Architecture of Supervisor Agent

3.3   The Model of iLearning System

The iLearning system is a hybrid system, which contains a multi-agent system, and two databases systems. The multi-agent system is composed by agents that cooperate between them. [2]

The agents have different goals and are managed by the supervisor agent.[9] The supervisor agent collaborates with the databases systems and coordinates the whole educational process. The model is presented in the figure 5.
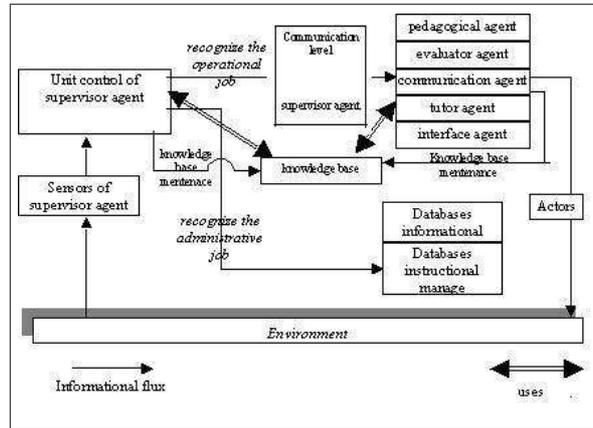

Figure 5: The Model of iLearning System

3.4   The iLearning's Programming and Reward Algorithm

To test the system, it was realized a software with Visual Studio 6.0 and Microsoft Agent Technology.

The algorithm used to learn the agents of iLearning system is a reward algorithm based on the Q-learning algorithm. This technique starts with an initial estimation $Q(s,a)$ for each pair state-action. When it is selected the action a in the state s, the system receives a reward $R(s,a)$ and it is observed the next state $s^{'}$. The Q-learning algorithm (Watkins, 1989) [11] appreciates the function value state-action as follows:

$$Q(s,a) = Q(s,a) + \alpha \times (R(s,a) + \gamma \times min_{a^{'}} Q(s^{'},a^{'}) - Q(s,a))$$

where $\alpha \in (0,1)$ is the learning rate, $\gamma \in (0,1)$ is the discount factor and $s^{'}$ is the state reached from state s executing the action a.[1] [5] The conceptual map of the course defines the space of system's state. A student with a certain learning style is with his studies in a node of the conceptual map and he was examined and received notes during the instruction process. The objective of the system is to maximize the results of students at different evaluations. The contribution of the authors is the adaptation of this algorithm for the pedagogical agents.

The reward is established through the student's evaluation in a node of the conceptual map and is defined according to the equation:

$$R(s,a) = note$$
or
$$R(s,a) = \frac{note \times p_{note} + apriorknowledge \times p_{base}}{time}$$

where: note is the score received by the student at his/her evaluation in a node of the conceptual map, *apriorknowledge* is the score received by students at the initial evaluation (before starting teaching the course if the student starts the course or the average of the scores received by the student at the evaluations in the nodes before the current node in the conceptual map), p_{note} and p_{base} are parameters.

The algorithm used is:

1. start with an array Q for all possible pairs state-action. Each item of the array is initialized with zero or a small value.

2. the optimal policy is initialized with a supervised policy. Q_{optim} is initialized with Q.

3. for each student the conceptual map is traversed and the Q array is calculated.

4. the Q is analyzed

4   The Experimental Results

To test the system there were selected students with different learning style (visual style, auditory style, kinesthetic style) and different ages and a module from the "Graph theory" course. The conceptual map [10] of the course with seven nodes is presented in the figure 6., each node has attached three pedagogical resource.

The software contains two agents: the supervisor agent and the pedagogical agent.[12] The pedagogical agent establishes the optimal pedagogical resources to teach the course.

The state of the system is defined by the combination (age, learning style and number of the node). The learning style no. 1 means the visual learning style, the style no. 2 means auditory learning style and the style no. 3 means the kinesthetic learning style. The age no. 1 means the persons with age between 20 and 30 years, the age no. 2 means the persons with age between 30 and 40 years and the age no. 3 means the persons with age between 40 and 50 years. The software teaches itself to teach better, selecting the best pedagogical resources for each node of the conceptual map.
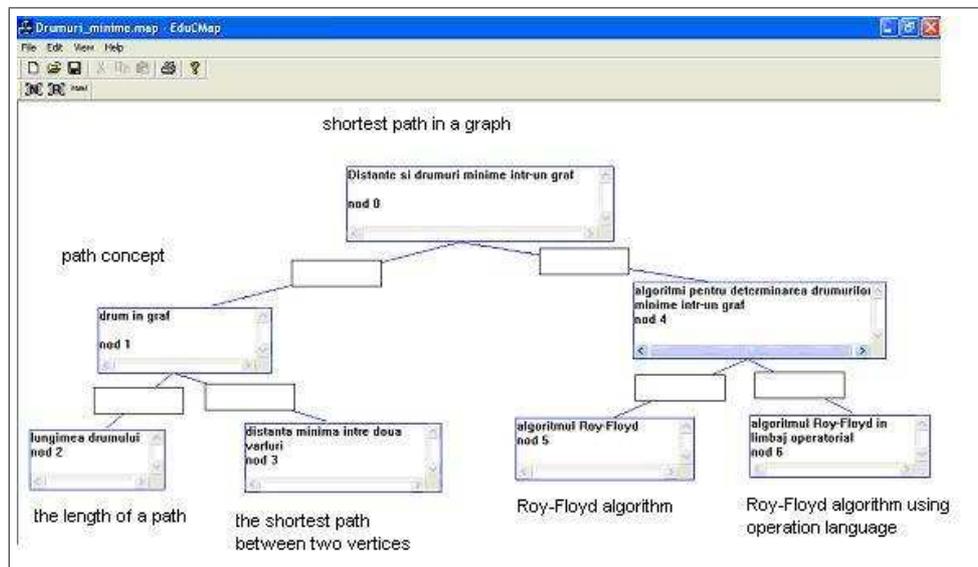
Figure 6: The conceptual map of the course Graph Theory, module Shortest Path

The results obtained for age category no. 2 and learning style no. 1 are presented in the table no. 1.

The last column contains the values for the parameter q from the reward algorithm.

The greater values of parameter q give the optimal policy (the best pedagogical resources for each node of the map according to the learning styles of students).

The optimal policy is described in the table no. 2.

The results obtained for age category no. 3 and learning style no. 1 are presented in the table no. 3.

Age	Learning style	No. node	No. of pedagogical resource	q
2	1	0	1	8.75
2	1	0	2	0
2	1	0	3	10.27
2	1	1	1	4
2	1	1	2	6
2	1	1	3	7
2	1	2	1	6.37
2	1	2	2	0
2	1	2	3	7.68
2	1	3	1	4.5
2	1	3	2	6.12
2	1	3	3	8.18
2	1	4	1	3.5
2	1	4	2	9.28
2	1	4	3	0
2	1	5	1	4
2	1	5	2	3.5
2	1	5	3	7
2	1	6	1	0
2	1	6	2	5.25
2	1	6	3	5.75

Table 1: Experimental Results for age 2 and learning style 1

Age	Learning style	No. node	No. of pedagogical resource
2	1	0	3
2	1	1	3
2	1	2	3
2	1	3	3
2	1	4	2
2	1	5	3
2	1	6	3

Table 2: The optimal policy for age 2 and learning style 1

Age	Learning style	No. node	No. of pedagogical resource	q
3	1	0	1	6.25
3	1	0	2	8.75
3	1	0	3	0
3	1	1	1	0
3	1	1	2	6.5
3	1	1	3	0
3	1	2	1	5
3	1	2	2	5.75
3	1	2	3	0
3	1	3	1	4
3	1	3	2	0
3	1	3	3	5
3	1	4	1	0
3	1	4	2	4.5
3	1	4	3	5
3	1	5	1	0
3	1	5	2	4.5
3	1	5	3	3.5
3	1	6	1	3.5
3	1	6	2	0
3	1	6	3	4.5

Table 3: Experimental Results for age 3 and learning style 1

The greater values of parameter q give the optimal policy (the best pedagogical resources for each node of map according to the learning styles of students). The optimal policy is described in the table no. 4.

It is true that the best results will be obtained if there are a lot of pedagogical resources for each node of the conceptual map. These pedagogical resources have to be in different formats (text, multimedia, audio, video) containing details, explanations, exercises, references, cases studies, and so forth.

There are a lot of software tools which can be used to make different files formats. Also, we need to improve the computers performances in order to use these kinds of softwares.

Age	Learning style	No. node	No. of pedagogical resource
3	1	0	2
3	1	1	2
3	1	2	2
3	1	3	3
3	1	4	3
3	1	5	2
3	1	6	3

Table 4: The optimal policy for age 3 and learning style 1

The teachers validate the results; the pedagogical resources no. 2 and no. 3 are the best pedagogical resources built for this course. The pedagogical resources no. 2 and 3 contain a lot of details, examples and explanations. The resource no. 3 (means the selection of the third pedagogical resource) is the best choice for persons with age group no. 2 and learning style no. 1. The resources no. 2 and 3 (means the

selection of the second and the third pedagogical resources) are the best choices for persons with age no. 3 and learning style no. 1.

5   Summary and Conclusions

The online instruction model can be implemented with the intelligent agent technology. The iLearning system proves this.

The system models the teaching-learning process so that it can adapt itself at the learning profile of each person. There are defined two kinds of agents: the execution agent and the supervisor agent. The author in the doctoral study tested the system and proved the importance of using the technology of intelligent agents in the online instruction systems.

References

[1] Bowling, M., Veloso M., Multiagent Learning Using a Variable Learning Rate, *Journal Artificial Intelligence*, Vol. 136, pp. 215-250, 2002.

[2] Buiu, C., Albu. M., *Agenti Software Inteligenti,* Editura ICPE, 2000.

[3] Gardner, H., *Intelligence Reframed, Multiple Intelligence for the 21st Century,* Published by Basic Books, 1999.

[4] Joyce, B, Weil. , Calhoun.E. , *Models of Teaching,* Published by Basic Books, 1999.

[5] Leon, F., Sova, I., Gâlea, D., Reinforcement Learning Strategies for Intelligent Agents in Knowledge-Based Information Systems, *Proceedings of the 8th International Symposium on Automatic Control and Computer Science*, Iaşi, 2004.

[6] Moise, G., The role of intelligent agents in online learning environment, *CBLIS Conference Proceedings*, pp. 98-105, 2005.

[7] Moise G., *A Software System For Online Learning Applied To Higher Education In The Field Of Computer Science,* Thesis: Petroleum-Gas University of Ploiesti, 2006.

[8] Müller, Jörg P., *The design of intelligent agents: a layered approach,* Lecture notes in computer science, Vol. 1177: Lecture notes in artificial intelligence, Springer-Verlag, 1996.

[9] Rao, A., S., Georgeff, M., P. , BDI Agents From Theory to Practice,*Proceedings of the First International Conference on Multi Agent Systems ICMAS-95*, San Francisco, 1995.

[10] Sowa, J.F. ,*Kowledge Representation Logical, Philosophical, and Computational Foundations,* Brooks/Cole, Thomson Learning.

[11] Watkins,C. , *Learning from Delayed Rewards,* Thesis: University of Cambridge, England.

[12] Wooldridge, M., Jennings, R., N. , BDI Agents From Theory to Practice,*Intelligent Agents: Theory and Practice, Knowledge Engineering Review* , Vol. 10 No 2, 1995.

[13] http://www.nwlink.com

Gabriela Moise
Petroleum-Gas University of Ploiesti
Computer Science Department
no. 39 Bd. Bucuresti, Ploiesti, Romania
E-mail: gmoise@upg-ploiesti.ro
Received: November 11, 2006

Editor's note about the author:

Gabriela Moise (born on February 13, 1969) graduated the Faculty of Mathematics, specialization Informatics of the Bucharest University. She worked as software development in the software industry. Since 2003 she is Lecturer at Petroleum-Gas University of Ploiesti. Her research fields are: e-learning, graph theory, intelligent agents, knowledge representation, e-health. She has (co)authored seven books and more than twenty research papers. She has participated to many international conferences in the e-learning and e-business area.