

Motivation and Evolutionary Pedagogical Agents

Emmanuel Blanchard

HERON Laboratory, Computer Sciences
Department, University of Montréal
blanchae@iro.umontreal.ca

Claude Frasson

HERON Laboratory, Computer Sciences
Department, University of Montréal
frasson@iro.umontreal.ca

Abstract: Motivating students is a major issue for current Intelligent Tutoring Systems. Modern theories of motivation (such as the Self Determination Theory) have shown the positive motivational impact of autonomy-support. Following this idea, we have proposed in a previous work an autonomy-supportive motivational design for eLearning systems. Later, the importance to take learner's culture into account appeared obvious to us if we wanted to correctly enhance/maintain the learner's motivation. Both these findings resulted in an ITS called MOCAS (Motivationally and Culturally Aware System). MOCAS is basically composed of a virtual world in which several pedagogical agents with different roles, behaviors and knowledge cooperate to provide a motivational (i.e. autonomy-supportive) and culturally adapted teaching to learners.

In the real world, pedagogical behaviors frequently determine whether a preceptor will be well accepted/respected or not by his/her pupils (and in many cases, this rating will be culturally dependant). But whatever preceptors are rated, they can not change from day to day what they truly are and how they behave. In Intelligent Tutoring Systems, pedagogical agents could have this ability to dynamically evolve. In this paper we define how to genetically adapt the crowd of pedagogical agents that is inside our MOCAS. This process allows the production of generations of pedagogical agents whose behaviors are more and more fitting the learners' motivational and cultural needs.

Keywords: Multi-Agents System, Pedagogical Agents, Genetic Evolution, Cultural Adaptation, Motivation.

INTRODUCTION

Motivating students is a major issue for current Intelligent Tutoring Systems [1, 7, 8, 21]. Modern theories of motivation (such as the Self Determination Theory [5, 13, 17, 22]) have shown the positive motivational impact of autonomy-support. Following this idea, we proposed in a previous work an autonomy-supportive motivational design for eLearning systems [1]. Later, the importance to take learner's culture into account appeared obvious to us if we wanted to correctly enhance/maintain the learner's motivation [2, 3]. Both these findings resulted in an ITS called MOCAS (Motivationally and Culturally Aware System). MOCAS is basically composed of a virtual world in which several pedagogical agents with different roles, behaviors and knowledge cooperate to provide a motivational (i.e. autonomy-supportive) and culturally adapted teaching to learners.

In the real world, pedagogical behaviors frequently determine whether a preceptor will be well accepted/respected or not by his/her pupils (and in many cases, this rating is obviously culturally-dependant). But whatever preceptors are rated, they can not change from day to day what they truly are and how they behave. In Intelligent Tutoring Systems, pedagogical agents could have this ability to dynamically evolve. In this paper, using genetic programming [16], we present a methodology for dynamically adapting the crowd of pedagogical agents that is inside MOCAS. This process allows producing agents whose behaviors are more and more fitting learners' motivational and cultural needs.

First, we present MOCAS and its two main concepts: its autonomy-supportive design and the rule-based methodology used to adapt its teaching given the cultural backgrounds of its learners. Then, we explain how to genetically represent our pedagogical agents and how we use a fitness value to select which agent profiles have to be suppressed and which ones have to be used during a reproduction process that results in a new generation of agents with original behaviors combinations. The fitness value itself is dynamically calculated during an auction-inspired protocol. This protocol uses the genetic representation of agents to select the one that will be in charge of teaching a specific part of the course to a learner.

OVERVIEW OF MOCAS: A MOTIVATIONAL AND CULTURALLY AWARE SYSTEM

A Motivation-Oriented System

One of the currently most popular approaches to deal with the motivation of people is to try to enhance the sense of autonomy people feel during an activity. Hence, according to a review of literature made by Reeve and his colleagues [22], many studies have shown that:

“Students with autonomy-supportive teachers compared to student with relatively controlling teachers, show greater mastery motivation, perceived competence and intrinsic motivation, greater conceptual understanding, higher academic performance, and greater persistence in school” [22].

Reeve has also succeeded in showing that increasing teachers’ autonomy support results in an enhancement of students’ engagement in a learning task.

Autonomy-support as a way of enhancing motivation is one of the major points of the Self Determination Theory (SDT, 5, 13, 17, 22) among others. The autonomy notion must be understood and used cautiously. In SDT, it totally differs from the notion of individualism [5]. In fact, the need for autonomy refers to the need for someone to see its behaviors as self-endorsed [17], meaning that the individual has taken the decision to do such behaviors because he expects that these behaviors will allow him to have positive results while performing an activity.

According to SDT, encouraging people to make choices during an activity is one of the methods used to provide autonomy. On the opposite, any action which controls or restrains someone’s behaviors has a negative impact on someone’s sense of autonomy, which results in lowering its motivation to persist in this activity. The design of MOCAS is aimed at providing autonomy to learners, which should lead to increased learners’ motivation.

In a previous work [1], we have argued that using an Open Virtual World is autonomy-supportive and, therefore, a well indicated design for supporting motivation. OVW is a frequent concept in the Role Playing Game field. MOCAS uses a 3D OVW but OVWs are not strictly related to 3D environments. The essence of OVW concerns the ability for the user to make choices while he is interacting with the environment, to be the initiator of his behaviors and not to have to follow a sequence of predefined interactions. To summarize, OVWs can be seen as discovery learning environment (environment where learners will try to “construct” the knowledge of the domain [26]) that are focused on ways of enhancing proactive behaviors of users.

Despite all positive motivational aspects of OVWs, a learner facing an environment where he is free to do many actions may sometimes forget his learning objectives. He may also decide not to do anything. Thus, even in an OVW, coaching the learner remains an important need in learning tasks. The objective for the system is to interact proactively towards the learner “only” when it is necessary.

Nick Jennings says that multi-agents systems can be used to present multiple perspectives of a problem [13] and, as we have seen before, autonomy-support can be done by offering multiple choices to a learner. Following this, we have inserted a set of pedagogical agents in our OVW. The aim of these agents is to provide to the learner, with different perspectives, the guidance/coaching necessary to keep him focused on the learning task and also on the learning contents. To this extent, our agents can have different roles (for example, in a medical simulation, roles could be doctor, nurse, patient...) and different attitudes. They are able to teach only concepts that fit with their roles (it means that they have access to all the resources related to theses concepts) but they can also refer the learner to another agent, the knowledge of which is better related to the needs of this learner. Furthermore, as noticed by Lester and his colleagues, such agents can “*play a critical motivational role as they interact with students*” and have “*an exceptionally positive impact on students*” [18]. Figure 1 presents the interface of a learner in MOCAS (the course here is concerning Greek mythology).

There are three different parts in this interface: *the interface for the game-like 3D environment* (where the avatar of the learner can navigate and interact with avatars of pedagogical agents), *the communication interface* (to allow the learner to communicate with other online learners) and *the learning content interface* (where information in relation with the domain to be learned will be culturally adapted). Then it can be displayed following different layouts given the recommendations of pedagogical agents). The learning content interface is a modular interface. For example, in figure 1, the loaded module displays an HTML file. But other modules exist that allow displaying video files, questions and answers interfaces... It is also possible to develop modules

specifically related to the domain to be learned. For example, we could imagine developing a command board that allows manipulating a 3D model of a medical device in a medical course.

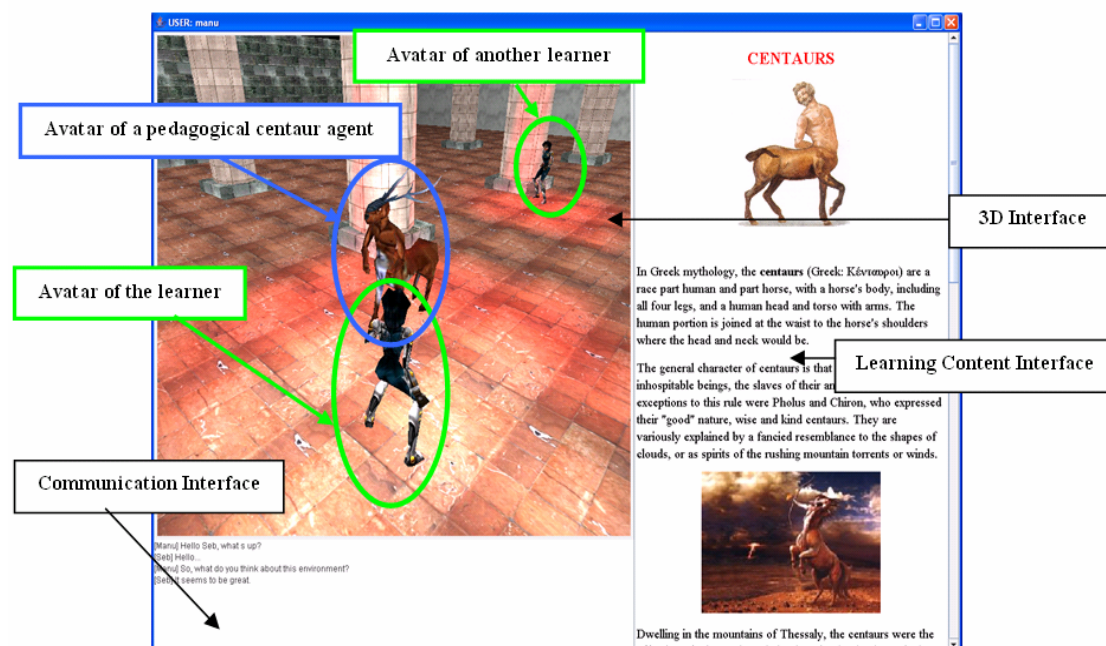


Figure 1: interface of the learner's application in MOCAS.

A Culturally Adaptive System

When reading cross-cultural literature, it appears clearly that ITS and eLearning in general can highly benefit from dealing with cultural differences. In fact, the cultural background affects many elements that are important in the ITS research field:

- *Emotions.* The frequency to which someone feels positive or negative emotions is culturally dependant [25], and the emotion categorization itself as positive or negative can in some cases depend on the cultural background [15]. We also believe that, depending on the culture, emotions of the learners can be expressed in a very different manner.
- *Preference for a pedagogical strategy.* For example, people from some cultures tend to prefer collaborative works whereas in other cultures, the preference is given to individualistic works [2].
- *Meanings given to concepts and symbols.* Many studies have shown that, depending on their cultural backgrounds, people can give drastically different meanings to concepts and symbols (see for example [12]).
- *Reward allocation.* Depending on the culture, the way teachers reward their learners and the ways these learners react to these rewards is different [11].
- *Test Anxiety.* Learners find more or less stressful to take a test depending on their culture [4].
- *Motivation and autonomy support.* Autonomy-support has been proven to be a cross-cultural way of enhancing motivation [13] but the methods used to fulfill this need can change from a culture to another [5, 17].

These examples stress the importance for ITS and eLearning systems in general to be aware of their learners' culture. According to this point, MOCAS tries to have some kind of *Cultural Intelligence*. This concept is defined by Earley and Mosakowski as a "*seemingly natural ability to interpret someone's unfamiliar and ambiguous gestures the way that person's compatriots would*" [9]. Inspired by this work, we think that a Culturally Intelligent System (such as MOCAS) must have two main capacities:

- *Understanding:* the ability for the system to interpret a learner's behavior/feeling/result depending on the learner's cultural specificities. This should allow the system to judge learners on the same basis also if they have different cultural backgrounds.

- *Adaptation*: the ability for the system to display different interfaces and/or to start different learning strategies depending on the learner's cultural specificities. This should allow the system to give a better answer to the needs of a learner.

In order to reach these objectives, we have developed a rule-based methodology that we first used to culturally adapt displayed learning contents [3]. It works as a recommendation system and we are currently enhancing it in order to be able to manage any kind of pedagogical resource.

Cultural rules are deduced from the cross-cultural research field. For example, the *Hofstede's system of values* [12], one of the major cross-cultural works of the twenty last years, supposes that national cultures exist and can be represented with five different dimensions. Values for these dimensions have been obtained for 50 different countries.

Among others things, our methodology can use Hofstede's dimensions and their values. It translates them as facts for a rule-based engine. It then processes these facts using different rules that we have defined in order to obtain attributes describing people of a given cultural group (for example, the *collaboration* attribute: people from such cultural group should prefer to work in a collaborative manner) and a degree of certainty for each of these attributes. To resume, a cultural group is described as a vector of couples "*attribute-weight*".

Each learner has a similar vector of couples which is initialized depending on the learner's cultural background. For each learner, a membership score is calculated for each cultural group (using the normalized distance between the vector of the learner and the vector of a given cultural group).

During the learning process, the weights in the learner's vector will evolve depending on his behaviors and learning successes/failures. The vector of each cultural group will also evolve depending on how its learners' vector evolve (the weight value of an attribute in the vector of a cultural group is the average of all the weights values for this attribute for all the members of this cultural group). It also results in the evolution of membership scores of the learner.

When a learner needs to learn some concept, pedagogical resources and strategies will be decided depending on the degrees of membership of this learner to different cultural groups. All the pedagogical resources and strategies are also dynamically rated in order to represent the interest to use them with learners of each cultural group.

REPRESENTING PEDAGOGICAL AGENTS IN A GENETIC MANNER

What is the reason of having agents with multiple behaviors in MOCAS?

As we have seen in the previous section, the way that learners interpret behaviors of their teachers can highly change depending on the culture among others things (see also [19]). Behaviors of a teacher strongly affect the receptivity of his learners during a learning session. In fact, learners continuously evaluate the attitude that a teacher provides to them in a specific situation.

We think that:

- This attitude is a combination of *multiple specialized behaviors*. For example, a teacher will use a different vocabulary, he will be more or less autonomy-supportive; he will activate different pedagogical strategies depending on the situation; he will have a specific manner to manage emotions of his learners; he will present information in different formats and so on.
- Some of these specialized behaviors may necessitate others behaviors, which results in complex interactions. For example, when a teacher wants to be autonomy-supportive, he may have to use a specific vocabulary depending on the learner's cultural background.

To synthesize, we can say that teachers can be differentiated by the behaviors they express. Such affirmation can also be applied to pedagogical agents in MOCAS (which are teachers in charge of transmitting the learning information) in order to offer multiple teaching attitudes. The choice among the different and original attitudes of pedagogical agents is another way of supporting the autonomy of the learner and de facto enhancing his motivation as seen before.

In MOCAS, we thus decided to assign different combinations of behaviors to our pedagogical agents in order to make each of our agents behave in an original manner. However, we need to find combinations of behaviors that correctly match to the learners' style. To this end, we have decided to use genetic programming. Some agent-based games already have explored a similar idea with great success (for example, *Creatures* [6], an adaptation of the game of life for general audience).

A quick overview of genetic programming

Genetic programming is an artificial intelligence methodology. It is frequently used for different kinds of applications such as (but not limited to) *Automatic Programming of Multi-Agent Systems*, *Evolution of Mental Models of the Environment*, *Optimization*, *Complex Adaptive Systems* (see [16] for further information). It follows the Darwinian principle of reproduction and survival of the best adapted to the environment. However, in genetic programming, the quality of the adaptation is evaluated depending on the fitness of the proposed solution to meet an objective of the system or to solve a problem. Genetic programming is based on a methodology where entities differ at the beginning of the process, which means they have different attributes i.e. genetic codes (and these attributes can trigger different behaviors). At a given moment, entities which best fit to the needs of the system are selected and used in a reproduction phase to produce new entities with original combinations of attributes: the genetic codes of two parents who are producing some of the currently best outcomes are mixed to produce two children with original genetic codes. It means that these children will have original combinations of behaviors that will be evaluated in their turn. Generation after generation, the objective is to provide a better answer to the needs of the system. In fact, thanks to this method, attitudes of all the pedagogical agents (i.e. their combinations of behaviors) will evolve in order to be better accepted by learners and also more useful for learning objectives.

Genetic representation in MOCAS

Before going further, we have to explain what we mean by *behavior* in our system. By behavior we define a set of actions (or minor behaviors) that are triggered in a determined manner (sequentially, in parallel, in loops...). As much as possible, this set of actions must be separated from resources. In fact, behaviors can be seen as templates that answer to the two following questions: *what to use* and/or *how to use it*? As we said before, we use the JADE platform to implement the behaviors of our pedagogical agents. We see three different types of behaviors:

- *Curriculum-dependent behaviors*: behaviors that can occur only in the context of a given course. For example a behavior designed to evaluate learners and which has been tailored for a geography course (by showing maps and asking which country it refers to) could not be useful in a math course. When creating a course, the author will decide if pedagogical agents in the course will be able to exhibit such behaviors or not.
- *Curriculum-independent behaviors*: behaviors that can occur in any kind of course. For example a behavior could evaluate learners by providing them with multiple choice questionnaires and multiple choice questionnaires are adaptable to any domain. Pedagogical agents will be able to exhibit these behaviors whatever the course will be.
- *Functional behaviors*: behaviors which are not directly related to the learning activity. For example, a behavior that allows a pedagogical agent to move all over the virtual world in order to reach a specific coordinate. These behaviors have the particularity to be similar for all the pedagogical agents. Thus, they are not taken into account in the process of genetic evolution that we describe below.

In MOCAS, we organize all the possible behaviors for pedagogical agents in classes of behaviors (for example emotional management, autonomy-supportive actions, politeness/vocabulary, pedagogical strategies, evaluation of the learner...). In a genetic metaphor, these classes can be seen as genes. Thus, each class/gene contains behaviors whose purpose is similar and these behaviors represent the alleles (i.e. the possible versions of the class/gene). Each of the behaviors is given a unique genetic code in this class (i.e. a string composed of 1 and 0).

Each pedagogical agent has a chromosome which enumerates the behaviors that this agent can exhibit. This chromosome is the concatenation of the genetic codes of behaviors taken from each class of behaviors. Figure 2 illustrates this initialization process for a pedagogical agent. In this example, a pedagogical agent will exhibit two different behaviors (one from a class A coded on 3 bits and one from a class B coded on 2 bits). The first initialization is a random process and here, behavior A3 has been chosen for class A and behavior B2 has been chosen for class B. It results in the 5 bits chromosome “011 10” that expresses that both these behaviors are exhibited by our agent until a possible recombination (we describe this process later).

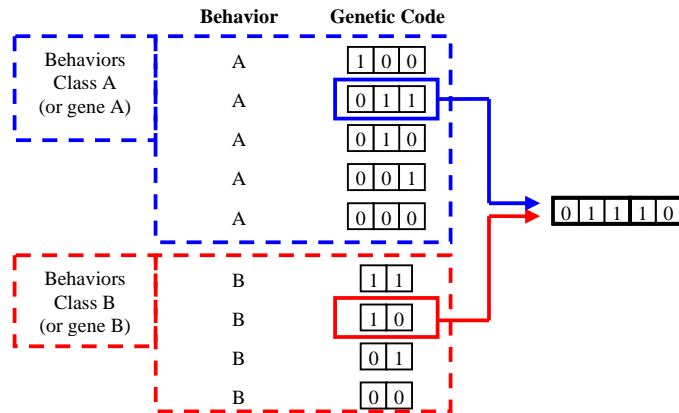


Figure 2: Genetic representation of a pedagogical agent in MOCAS.

The adaptation process in MOCAS

Many different ways exist on how to use genetic programming. Figure 3 describes the genetic process that we decided to use in order to obtain new generations of agents. It is divided in three phases that we call *initial phase*, *selection phase* and *reproduction phase*.

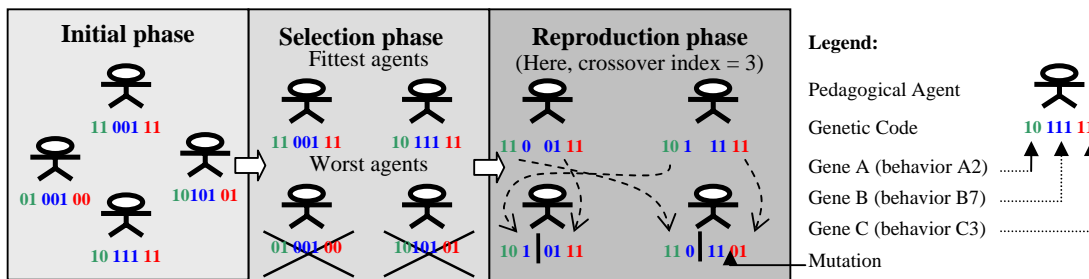


Figure 3: The process for the genetic generation of a new population of pedagogical agents in MOCAS.

Initial phase: the learning environment is populated with a set of pedagogical agents with different genetic codes (in the figure 3, a genetic code is composed of 3 genes A, B and C coded on 2, 3 and 2 bits respectively).

Selection phase: Frequently, all pedagogical agents will be rated depending on the satisfaction they give to the learner (see the description of our protocol below).

Reproduction phase: the best 50% agents (the fittest agents) remain similar and are also used to obtain new chromosomes for the worst agents. For this purpose, the fittest agents are coupled and a crossover index is chosen randomly for each couple (the crossover index is not necessary at the boundary of two different genes). Two new chromosomes are then obtained by inter-combining the right and left parts of the chromosomes of both parents. Sometimes, a mutation (i.e. the switch of a bit value) may occur but with a very low probability. This algorithm necessitates all possible genetic codes to be associated with a behavior. To follow this constraint, there are two possibilities:

- The number of possible behaviors for a gene/class is a power of 2.
- A wrong genetic code results in a “born-dead agent” and the process has to be repeated for this couple of parents until 2 corrects genetic codes are found.

Once two correct genetic codes (or chromosomes) are found (here “10 101 11” and “11 011 01”), they are allocated to agents that were low-rated. And a new process of rating begins for every pedagogical agent.

To carry out this process, it is obvious that agents in MOCAS have to cooperate. There are three kinds of agents in MOCAS:

- Multiple *pedagogical agents* (that we have described before).
- Multiple *learner’s agents*: those are interface agents and each of them is in charge of scrutinizing one learner’s interaction with the client interface. They also have the ability to modify the client interface (when requested by pedagogical agents).

- One *world agent*: in charge of monitoring the whole activity in a MOCAS world and determining global strategies of action for pedagogical agents. It also starts and manages the genetic evolution process after a determined period of time (or once a given number of human-agents interactions have occurred).

We are developing different protocols for agents teamwork based on the FIPA Agent Communication Language. One of these protocols is directly related to the adaptation process. It is described in the next part.

An auction-inspired protocol for the selection of pedagogical agents

As we have said before, pedagogical agents in MOCAS are in charge of providing the learning experience. Each of them is specialized and can teach only a specific part of the curriculum of this domain. When a pedagogical agent is asked by a learner to help him but it doesn't have the knowledge to help the learner, a protocol that we can view as a lightened version of the *Contract Net Interaction Protocol* specified by FIPA [10] is launched to find an agent that could help the learner. Figure 4 describes this protocol.

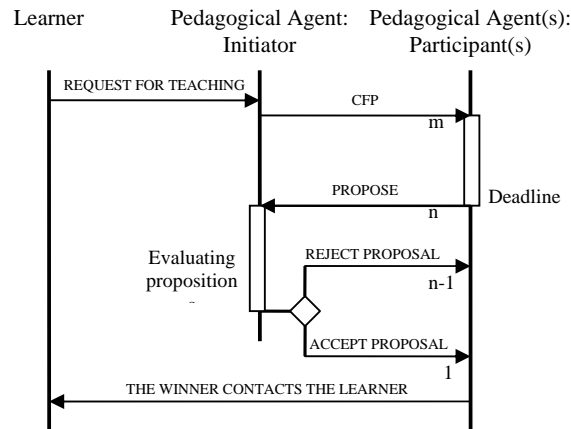


Figure 4: The protocol for the selection of a pedagogical agent in MOCAS.

When a learner requests a pedagogical agent to give him a teaching lesson that this agent is unable to give, the pedagogical agent initiates a multi-agents communication by sending a Call For Participants (CFP) to all the others pedagogical agents (m agents) which describes which teaching competence (which concepts) is asked for. Agents that can provide such teaching competence (n agents) send a proposition which contains their genetic code. As we have seen in the first part, all the behaviors are dynamically and culturally rated. The initiator agent will use such rating to decide, according to the learner cultural model, which pedagogical agent will be in charge of the teaching given the behaviors he can exhibit.

Once a pedagogical agent has given learning content related to learner' needs, all its behaviors are rated as if they were cultural resources [3]. Another rating also occurs. If the agent is found helpful/empathic for the learner (either by directly asking the learner or by analyzing his results), a fitness score attributed to each behavior he exhibited will be increased by one. Otherwise this score is decreased.

When the selection phase of the genetic process occurs (see the precedent part), the final fitness score for each pedagogical agent is determined by the world agent. This score is the sum of the fitness score of all the behaviors he can exhibit. The world agent determines which are the worst agent and attribute a new genetic code to them as presented in figure 3. Once this new generation of pedagogical agents is created, all the fitness scores of all the behaviors are reinitialized to 0 and a new rating process begins until it is decided by the world agent (after a given period of time) to create the next generation of pedagogical agents.

CONCLUSION AND FUTURE WORKS

In this paper we have presented a methodology that attributes genetic codes to all the pedagogical agents of MOCAS, our MOTivational and Culturally Aware System. The genetic code of a pedagogical agent determines which behaviors this agent can exhibit. We thus obtain agents that interact with learners in a different manner, depending on their global attitude (i.e. the combinations of their behaviors). According to the Self Determination Theory, this opportunity given to learners to choose among different perspectives of teaching is autonomy-supportive and autonomy-support positively affects the learner's motivation in a task.

Our genetic agents are also part of an evolutionary process which aims at creating original combinations of behaviors that could fulfill the needs of learners in a better manner than the previous generations. We mix the behaviors of agents whose global attitudes are highly rated according to a reproduction process. We thus obtain

these news combinations of behaviors and transfer them as new global attitudes to agents that were lowly rated. Then we restart evaluating the global attitudes of each agent until a new reproduction is decided. This is a Darwinian-like process that aims at encouraging the exhibition of behaviors that are highly profitable to learners.

The system prototype is active but we need to create more original behaviors for our agents in order to raise the number of possible combinations of behaviors and to be able to start a real cross-cultural test phase. In the purpose of structuring classes of behaviors, an ontology describing all the types of behaviors exhibited by human teachers could be very useful. The genetic code describing a behavior could also be upgraded in order to reflect the actions of this behavior. We are currently evaluating the possibility of creating persistent virtual learning environments. In the same way, we are working on a network topology based on a Meta server to allow multiple ITS to share their students models. It is clear to us that if there are more monitored students, the cultural adaptation should be more correct and the process of dynamic evolution should be more efficient

REFERENCES

1. Blanchard, E., Frasson, C. (2004). An autonomy-oriented system design for enhancement of Learner's motivation in eLearning. International Conference on Intelligent Tutoring Systems, Maceio, Brazil, pp. 45-54.
2. Blanchard, E., Frasson, C. (2005). Making Intelligent Tutoring Systems culturally aware: The use of Hofstede's cultural dimensions. International Conference on Artificial Intelligence, Las Vegas, pp. 644-649.
3. Blanchard, E., Razaki, R., Frasson, C. (2005). Cross-Cultural Adaptation of eLearning Contents: a Methodology. International Conference on E-Learning, Vancouver, Canada.
4. Cassady, J. C., Mohammed, A., Mathieu, L. (2004). Cross-cultural differences in test perceptions: Women in Kuwait and the United States. *Journal of Cross-cultural Psychology*, 35(6), pp. 713-718.
5. Chirkov, V. I., Ryan, R. M., Kim, Y., & Kaplan, U. (2003). Differentiating autonomy from individualism and independence: A self-determination theory perspective on internalization of cultural orientations and well-being. *Journal of Personality and Social Psychology*, 84(1), pp 97-110.
6. Creature Labs. http://www.gamewaredevelopment.co.uk/creatures_index.php
7. Del Soldato, T. (1994). Motivation in tutoring systems. Doctoral Thesis, Technical Report CSRP 303, School of Cognitive and Computing Sciences, University of Sussex.
8. De Vicente, A. (2003). Towards tutoring systems that detect students' motivation: an investigation. Doctoral Thesis. University of Edinburgh.
9. Earley, C., Mosakowski, E. (2004). Cultural Intelligence. *Harvard Business Review*, October 2004, pp. 139-146.
10. FIPA Contract Net Interaction Protocol Specification. <http://www.fipa.org/specs/fipa00029/SC00029H.pdf>
11. Fischer, R., Smith, P. (2003). Reward allocation and culture: a meta-analysis. *Journal of Cross-cultural Psychology*, 34(3), pp. 251-268.
12. Hofstede, G. (2001). *Culture's Consequences: Comparing Values, Behaviors, Institutions, and Organizations Across Nations*, 2nd edition, London: Sage.
13. Jennings N. R. (2001). An introduction to multi agents systems. Tutorial of International Conference on Autonomous Agents, Montréal.
14. Johnson, W. L., Vilhjálmsson, H., Marsella, S. (2004). Serious Games for Language Learning: How Much Game, How Much A.I.? International Conference on Artificial Intelligence in Education. Amsterdam, the Netherlands, pp 306-313.
15. Kim-Prieto, C., Fujita F., Diener E. (2004). Culture and structure of emotional experience. Unpublished Manuscript, University of Illinois, Urbana-Champaign.
16. Koza, J. R. (1998) Genetic programming. In Williams, James G. and Kent, Allen (eds). *Encyclopedia of Computer Science and Technology*. New York, NY: Marcel-Dekker. Vol 39. Supplement 24. pp 29 – 43.
17. Levesque, C., Zuehlke, A. N., Stanek, L. R., Ryan, R. M. (2004). Autonomy and Competence in German and American University Students: A Comparative Study Based on Self-Determination Theory. *Journal of Educational Psychology*, vol 96 (1). pp 68-84.
18. Lester, J. C., Converse, S.A., Kahler, S. E., Todd Barlow, S., Stone, B. A., Boghal, R, S. (1997). The Persona Effect: Affective Impact of Animated Pedagogical Agents. Conference of Human Factors in Computer Systems (CHI'97). Atlanta, GA. pp 359-366
19. Oishi, S. (2004). Personality in culture: A neo-Allportian view. *Journal of Research in Personality*. Vol 38(1), pp. 68-74.
20. Park, K., Leigh, J., Johnson, A. E., Carter B., Brody J., Sosnoski J. (2001). Distance learning classroom using Virtual Harlem, 7th International Conference on Virtual Systems and Multimedia. pp 489-498.
21. Qu, L., Johnson W.L. (2005). Detecting the Learner's Motivational States in an Interactive Learning Environment. International Conference on Artificial Intelligence in Education. Amsterdam, the Netherlands, pp 547-554.
22. Reeve, J., Jang, H., Carrell, D., Jeon. S., Barch, J. (2004). Enhancing Students' Engagement by Increasing Teachers' Autonomy Support. *Journal of Motivation and Emotion*, 28(2). pp 147-169.
23. Ryan, R. M. & Deci, E. L (2000). Self Determination Theory and the facilitation of intrinsic motivation, social development and well-being, *American Psychologist*, 55. pp 68-78.
24. Schank, R. C., Abelson R. P.(1977). *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*, L. Erlbaum, Hillsdale, NJ.
25. Scollon, C. N., Diener, E., Oishi, S., Biswas-Diener, R. (2004). Emotions across cultures and methods. *Journal of Cross-cultural Psychology*, 35(3). pp. 304-326.
26. Van Joolingen, W. (1999). Cognitive tools for discovery learning. *International Journal of Artificial Intelligence in Education*, 10. pp. 385-397.