

A Multiagent System to Support Problem-Based Learning

Laysa Mabel de O. Fontes¹, Francisco Milton Mendes Neto¹,
Alexandre A. A. Pontes²

¹Postgraduate Program in Computer Science, University of State of Rio Grande do Norte,
Federal University of Semi-Arid, Mossoró, Brazil;

²Superintendent of Information Technology and Communication, Federal University of Semi-Arid,
Mossoró, Brazil.

Email: {laysa, miltonmendes, alexandreadames}@ufersa.edu.br

Received October 15th, 2011; revised November 14th, 2011; accepted November 28th, 2011.

The problem-based learning (PBL) is a learning theory that emphasizes collaboration and teamwork to solve a problem. However, a problem that occurs frequently in the implementation of PBL is the out-of-context conversation, which is a situation in which the students lose focus and start talking about topics that are not related to the discussion. Another important aspect is related to the formation of groups in PBL. It might be difficult for the facilitator to assign students to groups at a distance, since the lack of presential contact makes it difficult to perceive important characteristics of the students' profiles involved in the process. Thus, this paper presents a Multiagent system to support PBL, with the objective of detecting and correcting problems inherent in the implementation of this learning theory.

Keywords: Problem-Based Learning, Multiagent System, Agents

Introduction

Distance Learning (DL) is a modality of teaching and learning that has grown and produced good results. Group activity is an important component in classroom teaching. Interactions between students over the course of some educational activity are crucial to the learning process, as each student shares with the others his/her knowledge, questions and impressions of what was discussed in class, thus enriching the learning process. This form of learning is called Collaborative Learning (Coutinho, 2007).

The computer support to collaborative learning is called Computer-Supported Collaborative Learning—CSCL (Dimitracopoulou, 2005). Systems that implement collaborative learning must include cooperation and communication mechanisms so that the learning process can be performed with quality. Also, such systems must have group creation mechanisms and ways of monitoring for the facilitators (teachers).

In the traditional approach, the teaching is teacher-centered. The foundation of said stance is that the teacher has the knowledge that must be transferred to the student. In this modality of teaching, classroom activities are developed under the exclusive direction of collective activity and the students are responsible for their learning. The discussion of ideas among members of the group increases interest and promotes critical reasoning.

The Problem-Based Learning (PBL), according to (Hmelo-Silver, 2004), is a method through which students learn while solving a problem that usually doesn't have a trivial solution and only one correct solution. The learning is student centric and self directed. The students, organized in small collaborative groups, work towards identifying what they must learn in order to solve the problem. The teacher acts as a facilitator in the learning process, instead of just transmitting knowledge (Fontes et al., 2010). In the collaborative learning approach, the students work together in small groups towards a common goal (Coutinho, 2007). Software agents have been used in an educational context. That technology has proven to be very promising

at aiding collaborative learning environments, boosting the process. They can be used, for instance, to support the fulfillment of a learning theory in a collaborative environment (Pontes, 2010).

A problem that occurs often in the process of applying PBL on distance learning is the dispersion of students. That is caused mainly by the physical absence of a teacher to direct the discussions.

Another important aspect in this context is group creation. On PBL, the members of a group are responsible for solving a problem and, in order to do that, they must have complementary competences regarding the matter. Also, it could be difficult for the facilitator to assign students to groups, since the lack of physical presence makes it difficult or him to know certain important features of the students involved in the process.

That way, aiming to solve the problems related to student dispersion and support group creation, a PBL-support multiagent system is presented.

This work is structured as follows: on Section 2 the main PBL-related concepts are described; Section 3 presents an explanation on multiagent systems; on Section 4, the agent-based approach to PBL support is presented; on Section 5, the related works are presented; at last, on Section 6, the final remarks and future work are presented.

Problem-Based Learning

The role of the facilitator is to guide students in this process, identifying possible deficiencies in their knowledge and skills necessary to solve the problem proposed. Thus, in this learning theory, rather than the facilitator simply transferring the knowledge to the students and then testing them through evaluations, he causes the students to apply their acquired knowledge in new situations. In this approach, students often face ill-structured problems and are motivated to discover, through investigation and research, useful solutions.

For successfully applying of PBL as pedagogic strategy the following stages must be accomplished (Hmelo-Silver, 2004): 1) the facilitator proposes an ill-structured problem to the students group; 2) the students try to generate facts and identify hypotheses about the problem through an initial brainstorming; 3) Then, the students formulate and analyze the problem aiming to generate ideas for problem solving; 4) After this, the students, supported by the facilitator, identify knowledge deficiencies for solving the problem by explanations and justifications; 5) In the following, the students look for new knowledge related to the domain, for following try to generate facts about this new knowledge; 6) at the end of each problem, the students reflect on the acquired knowledge. Figure 1 illustrates the PBL development cycle (Hmelo-Silver, 2004).

When applied, PBL offers some benefits, among which these stand out (Hmelo-Silver, 2004):

- Develops critical thinking and creativity on the student;
- Improves his ability to solve problems;
- Improves motivation;
- Helps students apply the knowledge they acquired to new situations.

Multiagent Systems

According to (Russell & Norving, 2002) agents are autonomous software entities that perceive their environment through sensors and perform actions on the environment through actuators, processing information and knowledge.

Multiagent systems can model complex system, allowing agents to have common or conflicting goals. Those agents can interact with each other in two ways: directly (via communication and negotiation) or indirectly (acting upon the environment). The agents can cooperate in order to achieve mutual benefits or compete to serve their own interests (Bellifemine, Caire, & Greenwood, 2007). Sensors are the agent's data inputs and the actuators are the ways through which the agent performs its actions and interacts with the environment.

A percept sequence is everything that has been picked up by the agent. The agent's behavior is defined by an agent function that maps any percept sequence to an action (Russell & Norving, 2002).

There are several types of agents. They can be of software or

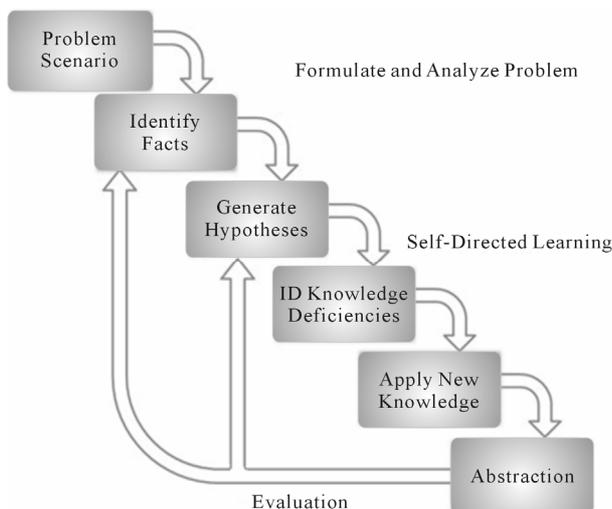


Figure 1.
PBL cycle.

of hardware, stationary or mobile, persistent or non-persistent, reactive or cognitive (intelligent). One of the most important classifications of agents is as reactive or cognitive.

Reactive agents are simple agents that note changes in the environment and react without any knowledge of previous actions. Since these agents have no memory, they are unable to plan future actions. Simple reactive agents select an action based on its current perception of the environment, ignoring previous perceptions. Cognitive agents are more complex because they have an explicit representation of both the environment and the other agents. This agent type has a memory, which enables it to plan future actions based on situations that took place previously (Russell & Norving, 2002).

A middle ground between the simple reactive agent and the cognitive agent is the reactive agent with internal state which, in order to achieve a more rational performance, have an internal state with aspects of the domain that may not be evident in the current perception. This state depends of previous perceptions of the environment, and is defined in a set of possible current internal states, $\Delta = \{\delta_1, \dots, \delta_l\}$.

This agent structure assumes that: 1) the agent receives information, through sensors, regarding the environment's state, defined in a set of possible states; 2) the agent has a perception sub-system and a decision-making subsystem; and 3) the agent executes the selected action on the environment through actuators (Pontes, 2010).

Agent Based Approach for Support PBL

Figure 2 presents the architecture proposed in this work.

The following subsections present the agent-based approaches to out of context conversation detection and group creation on PBL.

Agent Based Approach for Out-of-Context Conversations in PBL

Intelligent agents can perform many tasks in computer-supported collaborative learning, such as monitoring students' participation in discussions, facilitating the selection of topics for discussion, and assessing student performance in relation to the use of communication and cooperation tools available in the environment, among others. The use of agents to assist with these tasks is becoming increasingly important, mainly due to the increasing number of students who interact in learning support systems, which makes it very difficult to the facilitators to manage these activities at distance. The approach to out of context conversation detection is shown in Figure 3.

According to the approach presented in Figure 3, three types of agents are proposed: a Problem Detector Agent (PDAg), a Student Agent (SAG), and an Animated Interface Agent.

The Problem Detector Agent is responsible for detecting out-of-context conversations based on the environment's collaborative tools and an ontology. After detecting the students' focus has been lost, it notifies the Student Agent, which searches its history base to verify whether this student has already been stimulated.

If he has not been previously stimulated, the SAG will trigger the animated interface agent that will search for the first stimulus in the base of stimuli previously registered.

A stimulus is a text message shown by the interface agent. Then the animated interface agent will try to motivate the student with this stimulus. In the following, the interface agent will record the type of stimulus used in its historical base and

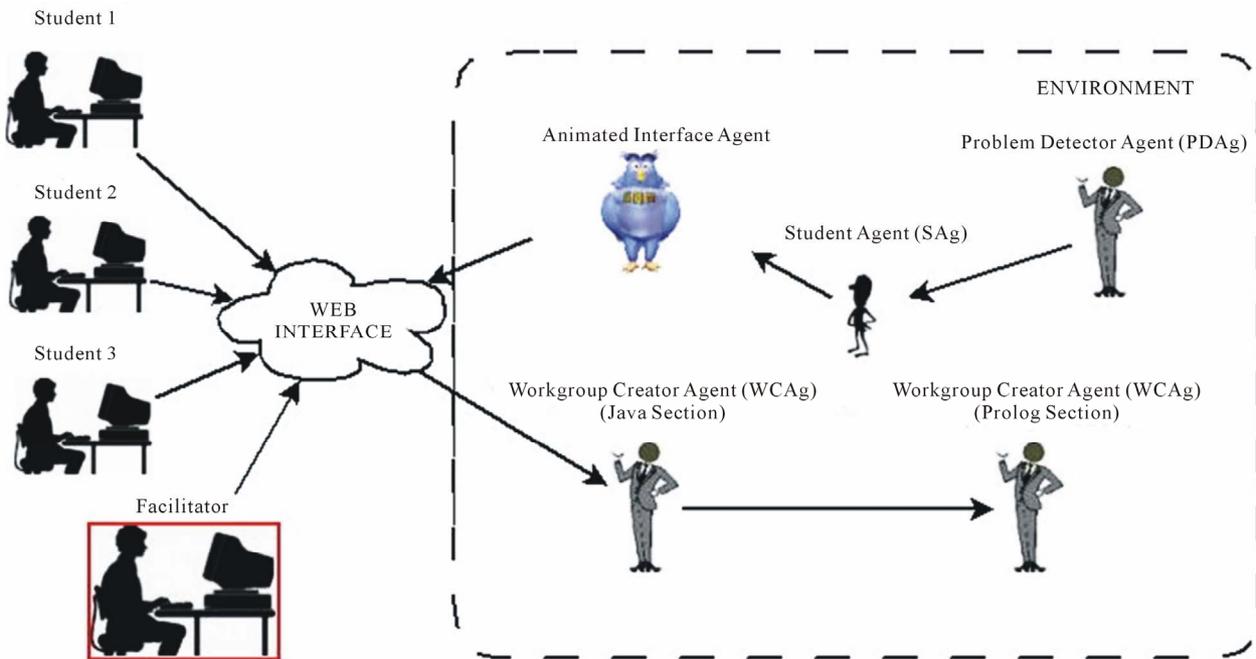


Figure 2. General architecture of agents.

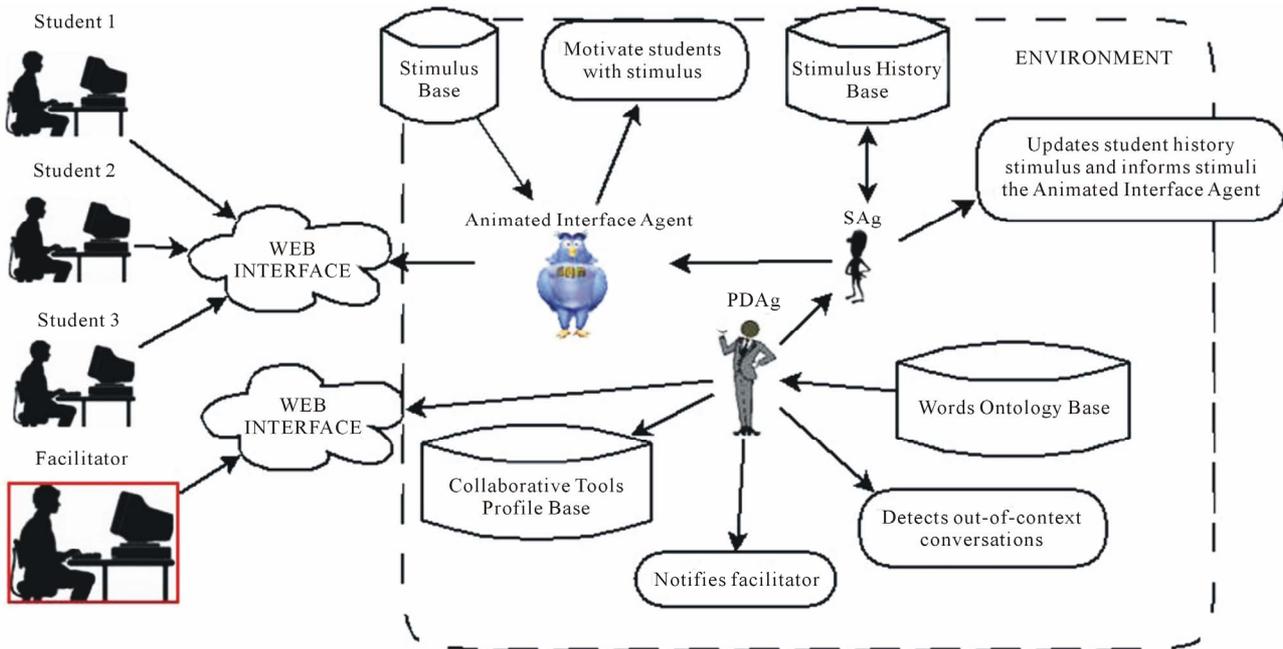


Figure 3. Agent based approach for out-of-context conversations in PBL.

notify the facilitator about the student’s situation.

However, if the SAg detecting in its historical base that the student was previously stimulated, it will check whether it has already passed the deadline (previously registered by the facilitator) given to the student improve their collaboration on the learning environment (ex. at the end of each day). If the deadline has not passed, it does nothing. But if the deadline has passed, it will notify the animated interface agent, which will search for the next stimulus in the base of stimuli by motivating the student in a different way. Then it records this information

in its historical base and it notifies the facilitator about the student’s situation, which is done through a message that is automatically sent via e-mail.

The agents will repeat this process while there is stimulus registered in the base of stimuli. The animated interface agent is responsible for motivating students to focus more of the discussions and to use the tools available in the virtual learning environment.

The PDAg is also responsible by detect students with unfocused behavior during the PBL’s interactions.

Agent Based Approach for Group Formation in PBL

The interaction in PBL plays a very important role on the learning process. In this context, the workgroups creation process in the learning environment is very important to the overall performance of the process. In presential learning, the students are very close to each other and, usually, the teacher knows each one of them, and the students know each other. In distance learning, the students are geographically distributed, therefore even the facilitator doesn't know all of the students, and the students don't know each other either. The facilitator must create the workgroups in PBL, but in distance learning he doesn't have enough information regarding the students in order to perform this task on his own. In the approach proposed in this paper, an agent is used to help the facilitator in this task. The workgroups creation process is illustrated in Figure 4.

The workgroups creation process happens as follows: the students, through a web-based interface, fill their profiles in at the beginning of the process. This process feeds a profile base that will be used in the workgroups creation process. The students' profile is composed of skills, acquisitions and deficiencies, where each of these has a level, which can be either low, medium or high. A student can have several skills, deficiencies and acquisitions.

The facilitator defines the profiles of the workgroups for each problem that must be solved through a web-based interface, similar to the one used by the student. The groups' profile is composed of skills, acquisitions and deficiencies, where each of these has a level, which can be either low, medium or high, and a fuzzy value that varies from 0.1 to 1. After the facilitator to create the groups' profiles, there will be a groups profile base that can be accessed by the WCAG. It is important to note that the desired profile, created by the facilitator, is what best matches the problem's resolution, and therefore, a student that has an approximate profile should have the competences needed to solve the proposed problem.

The WCAG is responsible for the automatic creation of groups. It was implemented using two programming languages: Java (Ken, 1996) and Prolog (Pereira, 2002). The Java section of the agent is responsible for the creation of candidates that are

apt to participate in a workgroup. This process is done by analyzing the students' profiles and the groups' profiles. After this analysis, it generates a file that will be the input for the Prolog section.

The generation of candidates is performed as follows: 1) the WCAG analyzes the profile of the group and checks whether there is a candidate that possesses some required skill; 2) if there is a candidate that has at least one required skill, this candidate is enclosed in a list of suitable candidates to compose the group. This process is done similarly for acquisitions and deficiencies. Thus, candidates that have at least one skill, acquisition or deficiency are included in the list of suitable candidates to join the group; 3) next, the WCAG generates an input for the section in Prolog. This input is called perception, and it is actually a text file that contains parameters that will be read by the agent session in Prolog and thus constitutes the interface between the two sections. An example of the file structure can be seen in Table 1.

The first parameter is the number of candidates. The second, a measure of similarity. The third is the amount of the universe of discourse. The fourth is the desired situation (sources and importance values) that represents the profile of the group and, finally, the fifth parameter is the current situation or the profile(s) of student(s) able to compose the group.

The Prolog section of the agent is responsible for the assignment of students to groups. At the end of this process, a file containing the workgroups is generated. The facilitator must analyze the result and decide whether he will accept the agent's suggestion.

Related Works

Multiagent Systems (MAS) have been widely used in educational applications. This technology has been quite promising as an aid in collaborative learning environments, making these environments more proactive and autonomous. MAS can be used, for example, to assist in the implementation of a particular learning theory in a collaborative environment.

In (Vizcaíno, 2005), a simulated student architecture is de-

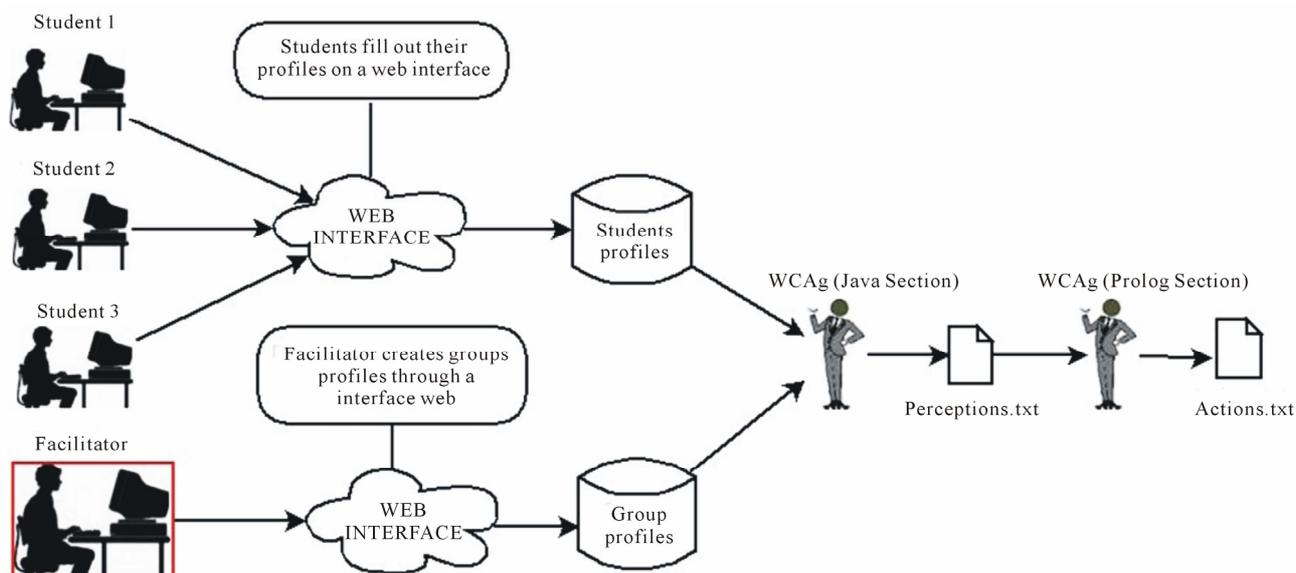


Figure 4.
Group formation process.

Table 1.
A file structure example.

1st Parameter	1
2nd Parameter	2
3rd Parameter	11
4th Parameter	[a, m, b, a, a] [0.8, 0.4, 0.2, 0.7, 1.0]
5th Parameter	[John', [m, m, b, m, b', 1]]

signed to detect and avoid three situations that decrease the benefits of learning in collaboration: off-topic conversations, students with passive behavior and problems related to students' learning.

As a distinctive feature of our work, we highlight the fact that our approach uses an animated interface agent with socio-affective features, i.e., when the problem detector agent identifies unfocused behavior, the animated interface agent tries to solve or minimize the problem by motivating the students to participate in activities and discussions. For this purpose, it uses text messages.

In (Moisil et al., 2006), a model for virtual learning environments is presented that employs intelligent agents to implement Vygotsky's sociocultural theory, focusing on the social aspect of interaction. The proposed model has several agents, among which we highlight the following: 1) the social agent, whose main goals are the construction of models for groups of students and the identification of groups of students that can cooperate in good conditions; 2) the tutor agent, which evaluates the student's educational goals and recommends some type of activity; and 3) the personal agents for assistance to the students, which monitor their activities and then inform other agents of the results of the monitoring.

As a distinctive feature of our work, we highlight the fact that our approach uses the PBL, which is a learning theory that has been proven to be effective (Tseng, Chiang, & Hsu, 2008; Strobel & Barneveld, 2009; Sendag & Ferhan, 2009).

In (Lima et al., 2005), the authors present an approach to group creation that is based on genetic algorithms, in which are applied the group's acceptance factors, by the teacher, taking into consideration the group's cohesion and the students' profiles, using socio-metric techniques. This approach is used on the NetClass cooperative learning environment.

In (Silveira & Barone, 2009), the authors apply multiagent techniques to collaborative groups creation in an Interactive Multiagent Environment for Learning on the Web. They present the definition and implementation of an agent architecture, modelled with genetic algorithms, as well as its integration with the TelEduc environment.

In (Felix & Tedesco, 2008), the Smart Chat Group tool is presented. It employs an intelligent agents society to create, suggest and monitor small learning groups based on the students' context's data.

As a distinctive feature of our work, we highlight the fact that the Group Creator Agent, proposed in this paper, performs the group creation based on both the students' and the groups' profiles, the latter having been created by the facilitator.

Final Remarks and Future Works

PBL is a learning theory that has been successfully applied to

virtual learning environments. This theory emphasizes teamwork and collaboration in order to solve a problem. However, a problem that often occurs is the dispersion of students during discussions on collaborative learning environments, which greatly influence their productivity. Another important problem regards the group creation process on PBL. It could be difficult for the teacher to assign students to groups without physical presence, since that lack of physical presence makes it difficult to perceive certain important features of the students involved in the process.

This paper presented an approach that uses software agents to avoid allowing the students to lose focus during interactions with other students and support group creation, providing the facilitator with support to solve these problems. Using the proposed approach, it is possible to achieve a reduction of student dispersion, as upon detecting the focus has been lost, it notifies the facilitator, who can take appropriate action. The architecture also provides support for group creation.

As the work's contribution, we highlight the development of an agent-based architecture that supports PBL on out of context conversation detection and group creation.

As future work, we intend to improve the out of context conversation detection approach and the group creation process presented in this paper, through literature study of other possible approaches to said problems. We also intend to approach other questions regarding PBL, as presented in (Pontes, 2010). Finally, we aim to perform a case study as a way to validate the solution presented in this work and a quantitative analysis in order to obtain statistical data that may verify the effectiveness of the proposed solution.

References

- Arnold, K., Gosling, J., & Holmes, D. (1996). *The Java™ Programming Language*. Upper Saddle River: Prentice Hall, 352.
- Bellifemine, F. L., Caire, G., & Greenwood, D. (2007). *Developing multiagent systems with JADE*. Hoboken: Wiley, 5.
[doi:10.1002/9780470058411](https://doi.org/10.1002/9780470058411)
- Coutinho, C. M. P., & Bottentuit, J. B. Jr. (2007). Collaborative learning using Wiki: A pilot study with master students in educational technology. *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications* (pp. 1786-1791). Vancouver.
- Dimitracopoulou, A. (2005). Designing collaborative learning systems: Current trends & future research agenda. *Proceedings of the 2005 Conference on Computer Support for Collaborative Learning: Learning 2005: The Next 10 Years!* (pp. 115-124). International Society of the Learning Sciences.
- Felix, Z. C., & Tedesco, P. A. (2008). Smart chat group: Ferramenta ciente de contexto para formação de grupos. Fortaleza: *Anais do Simpósio Brasileiro de Informática na Educação—SBIE, 19*, 643-654.
- Fontes, L. M. O., Mendes Neto, F. M., Pontes, A. A. A., & Campos, G. A. L. (2011). An agent-based architecture for supporting the work-groups creation and the detection of out-of-context conversation on problem-based learning in virtual learning environments. *Proceedings of the 2011 ACM Symposium on Applied Computing* (pp. 1175-1180). TaiChung, Taiwan: ACM SAC Proceedings. New York, NY: ACM.
- Hmelo-Silver, C. E. (2004). Problem-based learning: What and how do students learn? *Educational Psychology Review: 16*, 235-266.
[doi:10.1023/B:EDPR.0000034022.16470.f3](https://doi.org/10.1023/B:EDPR.0000034022.16470.f3)
- Lima, M. R. C., Labidi, S., Bastos Filho, O. C., & Fonseca, L. C. C. (2005). Aprendizagem cooperativa e o problema de formação de grupos. *Renote—Novas Tecnologias na Educação, 3*.
- Moisil, I., Pah, I., Barbat, B., & Popa, E. M. (2006). Socio-cultural modelling of the student as the main actor of a virtual learning environment. *Proceedings of the Wseas, International Conference on*

- Mathematical Methods and Computational Techniques in Electrical Engineering*, 8, Bucharest.
- Pereira, F. C. N., & Shieber, S. M. (2002). Prolog and natural-language analysis. *Microtome Publishing*, 64, 627-631.
- Pontes, A. A. A., Mendes Neto, F. M., & Campos, G. A. L. (2010). Multiagent system for detecting passive students in problem-based learning. *Advances in Soft Computing*, 71, 165-172. doi:10.1007/978-3-642-12433-4_20
- Russell, S., & Norving, P. (2002). Artificial intelligence: A modern approach. Upper Saddle River: Prentice Hall, 1132.
- Sendag, S., & Ferhan, O. H. (2009). Effects of an online problem based learning course on content knowledge acquisition and critical thinking skills. *Computers & Education: Elsevier*, 53, 132-141.
- Silveira, S. R., & Barone, D. A. C. (2009). Formação de grupos colaborativos em cursos a distância via web: Um estudo de caso utilizando técnicas de inteligência artificial. *Revista Brasileira de Informática na Educação*, 14.
- Strobel, J., & Van Barneveld, A. (2009). When is PBL more effective? A meta-synthesis of meta-analyses comparing PBL to conventional classrooms. *Interdisciplinary Journal of Problem-based Learning*, 3, 4.
- Tseng, K., Chiang, F., & Hsu, W. (2008). Interactive processes and learning attitudes in a web-based problem-based learning (PBL) platform. *Computers in Human Behavior: Elsevier*, 24, 940-955. doi:10.1016/j.chb.2007.02.023
- Vizcaíno, A. A. (2005). Simulated student can improve collaborative learning. *International Journal of Artificial Intelligence in Education*, 15, 3-40.