

A Computer-Supported Cooperative Learning System with Multiagent Intelligence

Leen-Kiat Soh, Nobel Khandaker, Xuliu Liu, and Hong Jiang

Computer Science and Engineering
University of Nebraska-Lincoln
256 Avery Hall, Lincoln, NE 66588-0115

1-402-472-6738

{lksoh, knobel, xuliu, jiang}@cse.unl.edu

ABSTRACT

In this paper, we describe an innovative infrastructure to support student participation and collaboration and help the instructor manage large or distance classrooms using multiagent system intelligence. The system, called I-MINDS, has a host of intelligent agents for each classroom: a teacher agent ranks and categorizes real-time questions from the students and collects statistics on student participation, a number of group agents that each maintains a collaborative group and facilitate student discussions, and a student agent for each student that profiles a student and finds compatible students to form the student's "buddy group". Each agent is capable of machine learning, thus improving its performance and services over time. These agents also interact and collaborate among themselves to exchange information and form coalitions dynamically to better serve the users. We have pilot-tested I-MINDS in GIS lectures, deployed I-MINDS in an introductory computer science course (CS1)'s laboratory, and evaluated the impact of I-MINDS based on student assessment. The results showed that students using I-MINDS performed (and outperformed in some aspects) as well as students in traditional settings.

Categories and Subject Descriptors

I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence – multiagent systems, intelligent agents.

General Terms

Algorithms, Design, Experimentation.

Keywords

Multiagent, Cooperative Learning, Distance Education.

1. INTRODUCTION

Students in large or distance classrooms often do not enjoy the same level and quality of student-to-student and student-to-instructor interactions during the class (synchronously) or after the class (asynchronously) as in smaller classrooms. Such interactions are, however, important to foster group-based learning and student affinity in a class. To improve such interactions, educators have turned to technology such as personal response systems, online chat rooms, teleconferencing tools, and computer-supported cooperative learning (CSCL) environments. However, these systems do not adapt to diverse student and instructor needs and behaviors. These systems are passive participants in the process and students and instructors often resort to other modes of interactions (e-mail, phone, or face-to-face meetings) in order to address individual problems. Further, the advent of Internet and multimedia technology has meant potentially drastic changes in the teaching and learning process from the traditional classroom setting to a more geographically distributed, virtual but still interactive one. Hence, there are both motivation and opportunity to develop intelligent CSCL systems.

The long-range goal of our research is to improve teaching and learning through technology, where intelligent agents actively support *cooperative learning* among students and support teachers in managing real-time classroom activities. A secondary goal is to bring the same quality of interaction and team building that invigorates the traditional classroom to distance education for both synchronous and asynchronous student learning. Note that "cooperative learning is instruction that involves students working in teams to accomplish a common goal" [4].

Our research has designed and developed an infrastructure called the Intelligent Multiagent Infrastructure for Distributed Systems in Education (or I-MINDS). I-MINDS consists of different intelligent agents working together to actively support student-student and student-instructor interactions. Specifically, there are agents that rank and categorize questions, and profile students to help the instructor manage a classroom in real-time; agents that find compatible agents to form "buddy groups" for students; and agents that form and maintain structured cooperative learning groups.

We have pilot tested I-MINDS and further deployed the software in the laboratory of an introductory computer science course at the Computer Science and Engineering Department of the University of Nebraska. Tests based on student assessments and control-

treatment groups showed that I-MINDS, though still needing better graphical user interfaces and further development, could support and improve student performance and instruction in several pedagogical aspects.

In the following, we first present I-MINDS' architecture in Section 2. Section 3 discusses I-MINDS' agents as well as their roles and relationships. We then describe the suite of features that allow the agents to actively support classroom interactions in Section 4. Section 5 gives an overview of the implementation of I-MINDS. In Section 6, we discuss the results of the pilot test and the deployment. Finally, we conclude with some future directions.

2. ARCHITECTURE

I-MINDS is built using a loosely-coupled layered architecture [7] (Figure 1), which makes I-MINDS flexible and usable in heterogeneous environments.

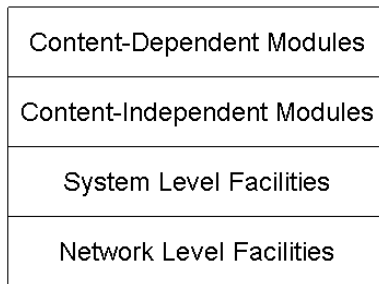


Figure 1: I-MINDS Agents and Topology

The bottom layer (i.e., the network layer) in I-MINDS provides the basic communication functionalities by using sockets. Since sockets are available in a variety of platforms across the network, this layer allows I-MINDS to communicate in heterogeneous situations.

The second layer provides system-level protocols and encapsulations with necessary abstraction to provide convenient communication and deployment functions to the upper layers. This level includes a Relational Database Management System (RDBMS)-type database (MySQL) and an audio/video server (Macromedia Flash[®] Communication Server). The database is used by the agents for fast storage and retrieval of information and the audio/video server helps establish audible and visual communication between the teacher and the students.

Finally the topmost two layers of I-MINDS contain the intelligent agents i.e., teacher agents, student agents, and group agents. Each intelligent agent has two sets of modules: content dependent and content independent. The content independent set provides the definitions and processes for general education-related services, while the content-dependent module handles specific course-related information and knowledge base, providing the required data and the heuristics used to gather, analyze, disseminate and process the generated data.

3. AGENTS and TOPOLOGY

Figure 2 shows an example of the topological infrastructure of I-MINDS [8]. The manager manages the ongoing classroom sessions. It manages system level information such as the list of

ongoing classes, list of courses and teachers, the login names and passwords for the students for each of the classrooms, etc. I-MINDS is capable of holding multiple concurrent classroom sessions. In each I-MINDS classroom session, the teacher, students and student groups are assigned the teacher agent, student agents and group agents, respectively. These agents support the person/group that they serve.

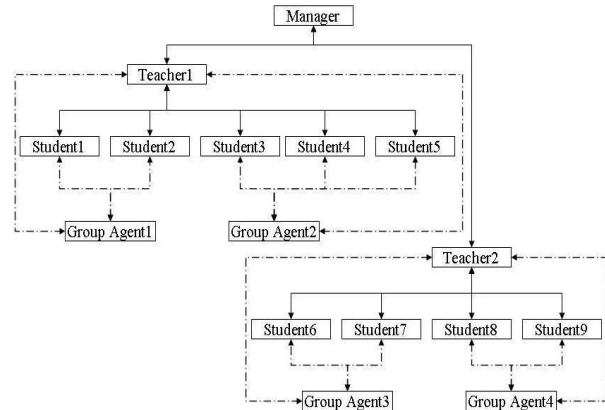


Figure 2: I-MINDS Agents and Topology

Intelligent agents are autonomous and can operate robustly in rapidly changing, unpredictable, or open environments. With these intelligent agents serving and catering to students' unique needs and behaviors, students will be able to participate in a computer-supported cooperative learning (CSCL) environment actively rather than listening to the lectures passively as in a traditional large or distance classroom. Currently, I-MINDS has three types of intelligent agents: (1) teacher agents, (2) student agents, and (3) group agents. A teacher agent, interacting with a teacher, is responsible for disseminating information streams to student agents, maintaining profiles for all students, assessing the progress and participation of different students, ranking and filtering of the questions asked by the students, and managing the progress of a classroom session. A student agent, on the other hand, mainly works as a personal helper to the student. The student agent manages the communication channels among students and between the teacher and the students. The student agent also presents the learning material to the student and forms coalitions with the other students for collaborative learning. The group agent forms and conducts structured cooperative learning such as the Jigsaw model [1], monitors and facilitates group activities.

3.1 Teacher Agent

In I-MINDS, a teacher agent has a graphical user interface (GUI) front-end, two sets of modules, and a database, as shown in Figure 3. A teacher agent interacts mainly with an instructor. It also sends instructions to the student agents, maintains student profiles, and receives questions from the student agents.

The GUI is equipped with various tools to create a classroom environment. For example, an instructor can capture and transmit desktop screenshots/slides to the class in real-time. An instructor can also communicate through the GUI with the students using text or audio/video using the message passing interface and video channel respectively [14].

The content-dependent module set of the teacher agent has a list of weighted keywords for a particular subject topic, rules and heuristics for ranking student questions for a particular course, and student profiles for a particular classroom. The content-independent module set of the teacher agent has a repository mechanism to allow the storage and access of the knowledge bases used in the content-dependent module set. It also has reinforcement learning mechanism, associated with question processing, to allow an instructor to implicitly “teach” the teacher agent how to rank questions better, as will be discussed in Section 4. There is also a Jigsaw support module that supports the structured cooperative learning Jigsaw model [1, 2], to be discussed as well in Section 4.

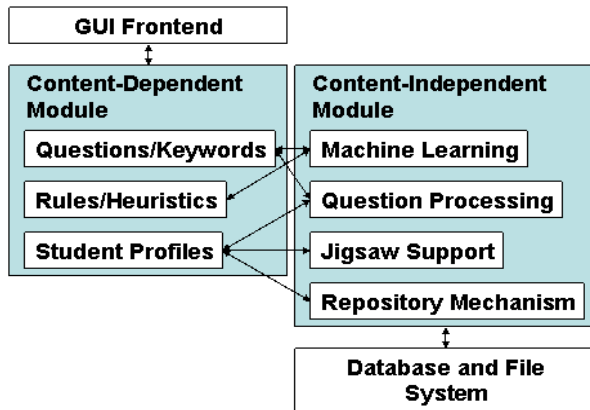


Figure 3: Teacher Agent

3.2 Student Agent

In an I-MINDS supported classroom, a student agent serves a unique student, and, similar to a teacher agent, it also has a GUI front-end, two module sets and a database as shown in Figure 4. A student agent primarily interacts with a student. It also exchanges information with the teacher agent and the group agents.

The student agent first allows for student login and course registration through communicating with the manager and the teacher agent, respectively. The student agent then acts as a medium: displaying the instructional materials received from the teacher agent to the student and relaying information from the student to the teacher agent.

In the content-dependent module, there are locally archived course materials and questions asked by the student and received from collaborative buddies [16]. Each student agent keeps a copy of profiles of all other students in the same class and synchronizes these profiles in real time with the teacher agent. The profiles are used as an important criterion by the student agent to automatically construct and refine the buddy groups. A student’s buddy group is one that the student relies on to collaborate in forums and “idea sketching”. This capability [15] can be turned on/off by the student.

When a student asks a question or performs some kind of collaborative activities (e.g., initiates a forum discussion or participates in sketching out ideas), his or her profile is changed and subsequently updated with the teacher agent and other student agents. The change might trigger other student agents to recruit

the student to join their buddy groups. Thus, a buddy group is dynamic. Further, if a student is not responsive to a buddy group that he or she is a member of, he or she can be dropped from the group. Hence, more responsive students will be approached more frequently than the others. The student agent also adjusts its heuristic rules according to the current classroom environment. For example, in a course in which students are very active in collaborative learning in every classroom session, a buddy who has not been active for the entire session may be dropped.

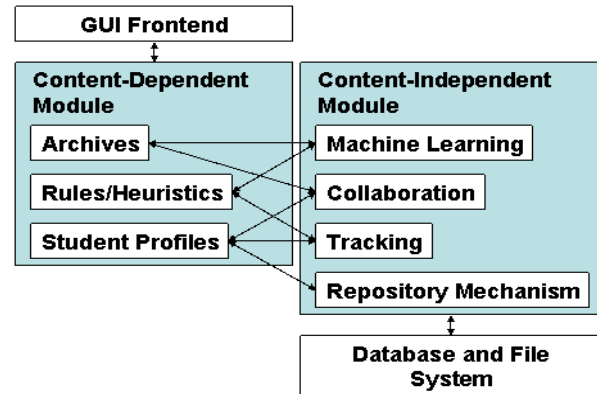


Figure 4: Student Agent

In the content-independent module, the tracking component tracks all activities and progresses of the student’s learning and collaboration. For example, I-MINDS provides a digital forum and a digital white board for students to communicate and collaborate. The digital white board facilitates “idea sketching”. All messages posted in the forum and activities on the digital white board are tracked and analyzed by the student agent. The tracking component also tracks the student’s attendance. If the student misses part of a class session, the student agent contacts the teacher agent, finds the archived course materials for the missed time period according to the timestamps and the syllabus, and reminds the student about the missed lectures. All the tracking results are sent to the teacher agent and factored into the student’s profile.

3.3 Group Agent

In I-MINDS, a group agent is activated when there are structured cooperative learning activities. Structured cooperative learning involves specified activities that explicitly require students to cooperate. Currently, I-MINDS implements the Jigsaw model [1], as will be discussed in Section 4. Since a group agent works behind the scene, it does not have a GUI front-end. Like the teacher and student agents, it has two content module sets and a database, as shown in Figure 5.

To support structured cooperative learning, the content-independent module set is capable of forming groups (or coalitions) of students based on their profiles, evaluating student perception of teamwork and other members through online surveys, tracking student participation, and improving its heuristics in monitoring and forming better groups. More will be discussed about the coalition formation module in Section 4.

The content-dependent module performs similar tasks as those discussed in previous sub-sections, archiving group activities (tasks and subtasks), and storing specific rules, heuristics and student profiles.

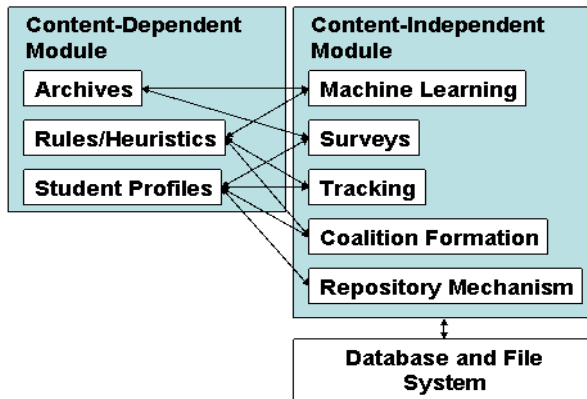


Figure 5: Group Agent

4. Intelligent and Multiagent Modules

In this section, we will discuss several key modules that makes our I-MINDS intelligent: automated question ranking that scores each question in real-time and ranks them, question grouping that classifies questions into similar groups based on their utterance classification and keywords, student profiling that keeps tracks of how each student performs in the classroom, cooperative student learning support implementing the Jigsaw process, and coalition formation that builds and refines the “buddy group” of each student.

4.1 Question Ranking

Question ranking is a key feature of the Question Processing module shown in Figure 2 that allows the teacher agent to manage large or distance classrooms for the instructor. When questions are asked, the teacher agent scores each question based on a set of keywords and heuristics. Keywords are subject-topic specific and weighted. Heuristics are course-specific. For example, a question asked by a student with a high-score profile will be given a high score; a question asked by a student who has never asked a question before will also be given a high score to encourage the student to ask more questions in the future; and a “why?” question is scored higher than a “what?” question. After scoring, the questions are ranked and displayed to the instructor. The instructor may choose to answer or discard a question. This is how the instructor’s actions trigger the reinforcement learning mechanism of the module. When the instructor chooses a lowly-ranked question over a highly-ranked question to answer, the teacher agent takes this as an indication that it has ranked the question incorrectly. It thus traces back to the heuristics and keywords that have contributed to the high score of the question and lower their weights; likewise, it increases the weights of the heuristics and keywords that have scored the question low in the first place. Thus, by answering and discarding ranked questions, the instructor implicitly teaches the teacher agent how to better score and rank the questions. Further, the Question Ranking feature also parses and extracts keywords from the questions. Questions that are answered trigger the learning of these

keywords. As a result, the teacher agent is able to add new keywords to the list of keywords, effectively building and refining important keywords for specific courses and subject topics. Details of this module can be found in [9].

4.2 Question Grouping

Question grouping is a key feature of the Question Processing module shown in Figure 2 that allows the teacher agent to address multiple similar questions together and profile students who ask similar questions. Future applications of this feature include automatically generated and answered Frequently Asked Questions (FAQs).

Question grouping is based on question classification and keyword matching. To achieve question classification, we adopt the *utterance classification* approach in the AutoTutor [3]. We use the ApplePie parser [11] and the utterance classifier program of AutoTutor [10] to classify a question into one of twelve classes such as Contribution, Discovery, and so on. This classification feature also allows the group agent to monitor the roles of group members in group discussions. Two questions are also considered similar if they share a high percentage of keywords.

4.3 Student Profiling

A student is profiled by I-MINDS’ agents in several ways. First, student-instructor interactions are profiled mainly based on the quality of questions that a student asks. Based on the grouping of these questions, students can also be profiled relative to others in terms of compatibility. Second, student-student interactions are profiled through the monitoring of a buddy group. This profiling process is carried out by a student agent from the viewpoint of the student it serves. That is, the student agent keeps a profile of its student’s buddies in terms of their responsiveness in collaborative activities (forums and digital whiteboard). These include the number of messages sent, type of messages, frequency of messages being sent, time spent on “idea sketching”, the number of times a student is dropped from a buddy group, and so on. Third, group interactions are monitored by the group agents during structured cooperative learning. These include the number of messages sent, types of messages, self-reported teamwork capabilities, peer-based evaluations as a team member, and so on. The teacher agent collects the student-student and group profiles from the student and group agents. The group agent does make use of the student-student profiles to help form coalitions.

4.4 Coalition Formation Module

To support the Jigsaw learning model we have designed an auction-based learning-enabled coalition formation algorithm called VALCAM that combines automatically tracked collaboration activities with subjective peer-based evaluations of a student’s performance in structured cooperative learning.

In VALCAM, the teacher agent acts as a coordinator or the coalition formation process and makes global decisions, such as, what should the least number of members in a coalition be, how long should the coalitions last, how the performance of each coalition should be evaluated, etc. The group agents then manage the coalitions. Each group agent monitors the performance and activities of the members of its assigned group. After the coalition has completed their tasks, the group agent also evaluates the performance of each student agent as a group member.

VALCAM algorithm is based on reinforcement learning of the student agents. In the beginning of every session the teacher agent chooses a few students and then initiates the group formation. During the group formation process, each student agent bids—an iterative Vickrey auction—to join its favorite group based on virtual currency earned from previous Jigsaw activities. Once the groups are formed, the student agents work with their group members to complete tasks assigned by the instructor. Finally, when the assigned task is completed, the student agents are rewarded with virtual currency based on their individual performance and their performance as a group member through surveys of their peers' evaluation of the student. Students who performed well will receive more virtual currency, allowing them to more successfully bid for their favorite groups in future sessions. Details of this algorithm can be found in [5].

5. REALIZATION of I-MINDS

We have implemented I-MINDS in Java (SDK 1.4.2). We have used Java's socket functionalities to establish communication among agents, Java's swing class to create interfaces. Furthermore, we have used Java's JDBC technologies to connect to our MySQL database repositories to store and retrieve all data. We have also used the Macromedia Flash[®] Communication Server to establish audio/visual link between the teacher and the students. Figures 6 and 7 show the interfaces of the student and teacher agents.

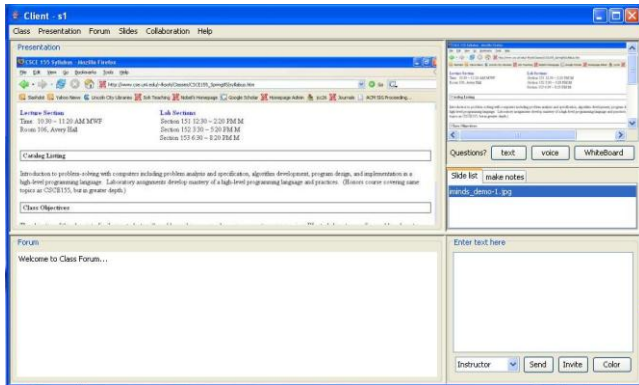


Figure 6: Student Agent



Figure 7: Teacher Agent

6. Results

6.1 Pilot Study

To determine the potential impact of I-MINDS on student learning, a pilot study [12] was conducted in May 2003 where the tool was used by subjects in a controlled experiment to assess what impact it had on student learning of Global Information Systems (GIS) content. GIS technology can be used for scientific investigations, resource management, and development planning.

In this study, all sessions were taught by the same instructor. There were 20 undergraduate and graduate student participants, divided into two groups. The *control* group was conducted like a traditional classroom, with the instructor and the students in the same room. The *treatment* group was conducted as a distance classroom where the instructor and the students were in separate rooms and their only communication channel was through I-MINDS.

All students took a pretest that would serve as a baseline score. Then, each group was given two 1-hour sessions. Session 1 discussed basic GIS topics and session 2 discussed advanced GIS topics. After each session, each student was required to take a post-test. The instructional content was the same for both groups.

Results from the two testing sessions were encouraging. Here we outline the results as reported previously in [12].

Although there was a slight difference in the means of the experiment and control groups for Test 2, this difference was not statistically significant ($p > .05$), and *the amount that the treatment group improved from the pretest to the posttest was nearly twice that of the control group.*

Comments from the university professor who used I-MINDS in teaching both of the content lessons were also encouraging. He indicated that the teaching tool was very easy to learn and use. He also said that the tool could enhance distance learning, especially by making it possible for building an archive of information that could be accessed “on-demand” by students. The instructor also noted that questions asked of him via I-MINDS tended to have higher quality, reflecting a deeper understanding, and demand a richer response than those questions posed during the control sessions.

Details of this pilot study can be found in [12].

6.2 Deployment with Jigsaw

In Spring 2005, we deployed I-MINDS in the laboratory of an introductory computer science course at the University of Nebraska. The course is CS1, the first core course for computer science and computer engineering majors in the Computer Science and Engineering Department. The course had three 1-hour weekly lectures and one 2-hour weekly laboratory sessions. In each lab session, students were given specific lab activities to experiment with Java and practice hands-on to solve CS problems. In Spring 2005, there were three different lab sections, each with about 20 students.

Our study utilized two sections. In the control section, students worked in Jigsaw cooperative learning groups without using I-MINDS. Students were allowed to move around in the room to join their Jigsaw groups to carry out face-to-face discussions. In the treatment section, students worked in Jigsaw cooperative learning groups using I-MINDS. Students were told to stay at

their computers and were only allowed to communicate via I-MINDS. With this setup, we essentially simulated a distance classroom environment.

For each lab, the students were given a lab handout with a list of activities—thus, a lab is a task and its activities are the subtasks. We conducted the study for three weeks, covering topics in debugging and testing, Unified Modeling Language (UML), and recursion.

The students of both the control group and treatment group were required to complete the tasks and subtasks in the four Jigsaw Phases. Table 1 shows the four Jigsaw phases with the allocated amount of time.

Table 1: Time allocations of Jigsaw phases

Jigsaw Phase	Time (min)
Introduction	15
Focused Exploration	30
Reporting and Reshaping	20
Integration and Evaluation	20

In each section, the instructor announced the main groups. In the control section, this was done manually. In the treatment section, I-MINDS automatically performed this using its VALCAM algorithm (Section 4.5). Once the main groups were formed, the teacher agent formed the focus groups by randomly selecting students from the main group. After the focus groups were formed, every focus group was assigned one subtask randomly. After the subtask assignment the focused exploration phase was started. In this phase the three Jigsaw Phases were carried out in order. During these three phases the student agents and the group agents monitored and controlled the activities of the students and the student groups respectively. After the three Jigsaw Phases were executed, all the students filled out the Peer Rating Questionnaire and Team Based Efficacy Questionnaire and took a 10-minute post-test. This 10-minute post-test score was used as the measure of student performance in terms of understanding the topic of the lab. Details of this process can be found in [13].

Table 2 shows the post-test scores of the three sessions for the “Control” (i.e., Jigsaw without I-MINDS) and “Treatment” (i.e., Jigsaw with I-MINDS) sections. The results indicate that students using I-MINDS for the Jigsaw activities were able to obtain comparable and even better post-test scores. We had initially hypothesized that the students in the control section would perform better than the students in the treatment section simply because the students in the former would have a chance to discuss face-to-face and were able to use paper and pencil to draw and share the same computer screen during the stages 2 and 3 of the Jigsaw process. On the other hand, I-MINDS, still lacking user-friendly GUI, had been expected to hinder such free, natural interactions among students, leading to ineffective collaborations. Thus, the result had not been expected. Students in the Treatment section performed better than the students in the Control section in the later two sessions. Further, students in the Treatment section also achieved better standard deviation—meaning that these students’ post-test scores were more tightly clustered than those of the Control section. Upon closer analysis, we suspect

that the act of typing and communicating through the forum and digital whiteboard of I-MINDS forced the students to articulate explicitly their thoughts and focused their attention to the tasks at hand. This in turn improved student understanding of the subject matter.

The results were very encouraging: without face-to-face interactions, students carried out their I-MINDS-supported Jigsaw tasks and performed relatively well in individual post-tests.

Table 2: Student performance for the Control (Jigsaw only) and Treatment (Jigsaw w. I-MINDS) sections

Peer Rating Questionnaire Results				
Week	Control Section		Treatment Section	
	Mean	Std. Dev	Mean	Std. Dev
1	42.02	2.68	35.60	5.97
2	39.06	7.05	33.30	8.65
3	39.90	4.80	38.59	6.34

Next, we look at the surveys in terms of peer rating and team-based efficacy. These surveys were conducted in both sections after each lab session. Tables 3 and 4 show the results. The *peer rating* average is the average peer rating scores that each student gave to his or her group members. As evidenced in the scores, students in the Control section rated their peers better than the students in the Treatment section. This is possibly due to the face-to-face interaction. After all, students interacting through I-MINDS could not enjoy the advantages of face-to-face interactions such as facial expressions, the spontaneous free-flowing of ideas, and more immediate feedback in their discussions. This observation indicates that I-MINDS still lacks sufficient graphical user interface and multimedia capabilities to fully capture real-time characteristics.

The team-based efficacy was collected after each lab based on a set of questions. It measured how a student viewed how well its group had performed. Once again, students in the Control section approved of their team-based activities more than the students in the Treatment section. Since there was no difference in the scores in tasks for the Control and Treatment sections, and that students in the Treatment section had more similar post-test scores, that means this perception had been likely biased by the ease-of-use (or the lack thereof) of I-MINDS.

Table 3: Peer Rating Questionnaire Results

Peer Rating Questionnaire Results				
Week	Control Section		Treatment Section	
	Mean	Std. Dev	Mean	Std. Dev
1	42.02	2.68	32.45	5.78
2	39.06	7.05	37.72	4.60
3	39.90	4.80	34.63	8.08

Table 4: Team Based Efficacy Questionnaire Results

Team Based Efficacy Questionnaire Results				
Week	Control Section		Treatment Section	
	Mean	Std. Dev	Mean	Std. Dev
1	29.87	2.51	27.72	5.08
2	31.86	3.48	29.20	2.78
3	30.08	3.02	28.25	4.02

Looking at the data tracked by the I-MINDS agents and comparing them to post-test scores, we observe the following. First, students rate their peers subjectively: average peer rating that a student gave to his or her peers is negatively correlated to his or her post test score with a correlation of -0.63. Second, students who do well in the post-tests tend to send long messages with a correlation of 0.51.

7. CONCLUSION

We have described a computer-supported cooperative learning system called I-MINDS that uses multiagent intelligence. We have discussed the structure and functionalities of the different agents in I-MINDS. Then we have described the intelligent and multiagent modules that allow the teacher, group, and student agents to collaborate to actively support student-student and student-instructor interactions. To demonstrate the effectiveness of I-MINDS, we have presented the results of our pilot study and also the results of incorporating Jigsaw Framework in I-MINDS in real-time classroom situations.

The results of the pilot study shows that the instructor and the students can both use I-MINDS effectively as a framework for delivering distance education.

Furthermore, the results of our CS1 lab deployment shows, although face-to-face activities were perceived to be better by the students, I-MINDS-supported activities were able to produce better or comparable post-test scores and were able to yield more similar scores among the students.

Future work includes continued deployment of I-MINDS in the classroom, improvement of the GUI front-end for the teacher and student agents to better support collaborative, real-time programming, and extension of I-MINDS to work more seamlessly with existing Interactive Design Environments (IDEs). Additional research includes the refinement of the VALCAM algorithm to more fairly synthesize subjective peer-based evaluation and agent-tracked empirical data in determining the amount of virtual currency rewarded to each student.

8. ACKNOWLEDGMENTS

The I-MINDS project was first developed with a seed grant from the National Center for Information Technology in Education (NCITE). It was later sponsored partially through an NSF SBIR grant DMI-044129. The authors would like to thank Phanivas

Vemuri, Suresh Namala, and Xuesong Zhang for their programming work.

9. REFERENCES

- [1] Aronson, E., Blaney, N., Sikes, J., Stephan C., and Snapp, M. *The Jigsaw Classroom*. Beverly Hills, CA: Sage, 1978.
- [2] Clarke, J. *Pieces of the Puzzle: The Jigsaw Method*. In *Handbook of Cooperative Learning Methods*. Westport, CT: Greenwood Press, 1994, 34-50.
- [3] Graesser, A. C., Wiemer-Hastings, K., Wiemer-Hastings, P., Kreuz, R. and Tutoring Research Group. Autotutor: A simulation of a human tutor. In *J. Cognitive Systems Research*, 1999, vol. 1, 35-51.
- [4] Johnson, D. W., Johnson, R. T. and Smith, K. A. Cooperative learning: increasing college faculty instructional productivity. In *ASHE-ERIC Higher Education Report No. 4*, George Washington University, 1991.
- [5] Khandaker, N. VALCAM - an auction based learning enabled multiagent coalition formation algorithm for real-world applications. M.S. Thesis, Univ. of Nebraska, NE, 2005.
- [6] Khandaker, N. Question classification and question similarity in I-MINDS. In *Independent Research Report*, Dept. Computer Science & Engineering, Univ. Nebraska, Lincoln, NE, 2004.
- [7] Liu, X., Zhang, X., Al-Jaroodi, J., Vemuri, P., Jiang, H. and Soh, L.-K. I-MINDS: An Application of Multiagent System Intelligence to On-Line Education. In *Proc. IEEE-SMC*, 2003, 4864-4871.
- [8] Liu, X., Zhang, X., Soh, L.-K., Al-Jaroodi, J. and Jiang, H. A Distributed, Multiagent Infrastructure for Real-Time, Virtual Classrooms. In *Proc. ICCE*, 2003, 640-647.
- [9] Namala, S. An intelligent module for I-MINDS. M.S. Project Report, CSE Dept. Univ. of Nebraska, NE, 2004.
- [10] Olney, A., Louwerse, M. Mathews, E., Marineau, J., Hite-Mitchell, H. and Graesser, A. Utterance classification in Autotutor. In *Proc. HLT-NAACL Workshop*, 2003, 1-8.
- [11] Sekine, S. and Grisman, R. A corpus-based probabilistic grammar with only two non-terminals. In *Proc. IWPT*, 1995, 216-223.
- [12] Soh, L.-K., Jiang, H. and Ansoorge, C. Agent-Based Cooperative Learning: A Proof-of-Concept Experiment. In *Proc. SIGCSE*, 2004, 368-372.
- [13] Soh, L.-K., Khandaker, N., Liu X. and Jiang, H. Computer-Supported structured cooperative learning. To appear in *ICCE'2005*, 2005.
- [14] Vemuri, P. Multimedia Support for I-MINDS. M.S. Project Report. Dept. Computer Science & Engineering, Univ. Nebraska, Lincoln, NE, 2003.
- [15] Ye, Y. and Churchill, E. *Agent Supported Cooperative Work*. Boston, MA: Kluwer Academic Publishers, 2003.
- [16] Zhang, X. I-MINDS: An Intelligent Multiagent System Supported Teaching and Cooperative Learning Environment. M.S. Thesis, Dept. Computer Science & Engineering, Univ. Nebraska, Lincoln, NE, 2004.