

Multiagent Coalition Formation for Computer-Supported Cooperative Learning

Leen-Kiat Soh, Nobel Khandaker and Hong Jiang

Computer Science and Engineering
University of Nebraska-Lincoln
256 Avery Hall, Lincoln, NE 66588-0115
1-402-472-6738
{lksoh, knobel, jiang}@cse.unl.edu

Abstract

In this paper, we describe a computer-supported cooperative learning system in education and the results of its deployment. The system, called I-MINDS, consists of a set of teacher agents, group agents, and student agents. While the agents possess individual intelligent capabilities, the novel invention of I-MINDS lies in multiagent intelligence and coalition formation. I-MINDS supports student participation and collaboration and helps the instructor manage large, distance classrooms. Specifically, it uses a Vickrey auction-based and learning-enabled algorithm called VALCAM to form student groups in a structured cooperative learning setting. We have deployed I-MINDS in an introductory computer science course (CS1) and conducted experiments in the Spring and Fall semesters of 2005 to study how I-MINDS-supported collaboration fares against traditional, face-to-face collaboration. Results showed that students using I-MINDS performed (and outperformed in some aspects) as well as students in traditional settings.

Introduction

Computer-supported cooperative learning (CSCL) systems are becoming more popular to support large or distance classrooms, to facilitate student-to-student and student-to-instructor interactions synchronously or asynchronously. However, these CSCL systems (e.g., Caverly and MacDonald 2002; Betbeder et al. 2003; Chan et al. 2003; Chang et al. 2003; Salcedo et al. 2003; Sridharan et al. 2003) do not adapt to diverse student and instructor needs and behaviors. Instead, these systems are usually passive participants in the process and students and instructors often resort to other modes of external interactions (e-mail, phone, or face-to-face meetings) in order to address individual problems. An instructor would have to monitor student activities and be required to process what he or she observes in order to generate the proper response without intelligent help from the CSCL system; while a student or a group of students would have to seek out their compatible peers to work together and to maintain their own peer

groups. These activities are difficult to achieve in large or distance classrooms where an instructor does not necessarily have the luxury to observe student activities accurately and nor does a student. Thus, there is a need for intelligent CSCL systems that could actively facilitate the above activities.

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. These agents work together to actively support student-student and student-instructor interactions in a typical classroom in the distance education setting. Specifically, there are agents that rank and categorize questions, and profile students to help the instructor manage a classroom in real-time; and agents that form and maintain structured cooperative learning groups. Powering the I-MINDS is a multiagent coalition formation algorithm specially designed to help form student groups based on student performances over time. This algorithm, called VALCAM, is based on a Vickrey auction setting and employs peer evaluation and profiling to improve the student groups.

We have deployed I-MINDS in an introductory computer science course (CS1) at the Department of Computer Science and Engineering of the University of Nebraska for the past year (2 semesters). Specifically, we have used a control-treatment protocol to study the use of I-MINDS in place of face-to-face collaboration among students in weekly laboratory sessions. Results have shown that I-MINDS, though still needing better graphical user interfaces and further development, could support cooperative learning effectively.

In the following, we first present I-MINDS agents, briefly discussing the intelligent modules of these agents to support cooperative learning. Then we describe the VALCAM-based coalition formation module. Here we also present the structured cooperative learning Jigsaw model that has been incorporated into I-MINDS. Subsequently, we offer some implementation and architectural details. We then discuss the deployment results of our CSCL application. Finally, we conclude with some future directions.

Agents and Topology

I-MINDS consists of three types of agents: teacher agents, group agents, and student agents. Figure 1 shows an example of the topological infrastructure of I-MINDS (Liu et al. 2003b). The manager, simply an administrative module, manages the ongoing classroom sessions and 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. In an I-MINDS classroom session, the teacher, students and student groups are assigned a teacher agent, student agents and group agents, respectively. Each of these agents supports the user or group to which it has been assigned.

Briefly, 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 a specific student. The student agent also presents the learning material to the student and forms coalitions with other students for collaborative learning. A group agent forms and conducts structured cooperative learning such as the Jigsaw model (Clarke 1994), monitors and facilitates group activities. In the following, we highlight some of the intelligent modules within each agent. For details, readers are referred to (Soh et al. 2006).

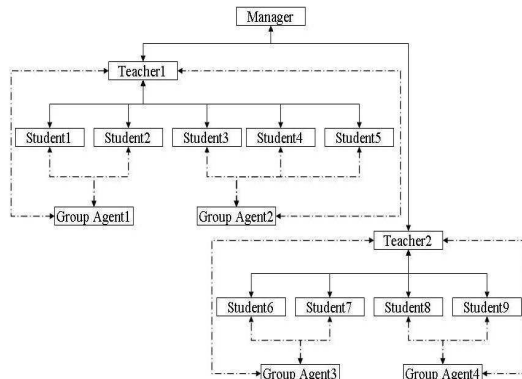


Figure 1: I-MINDS Agents and Topology

Teacher Agent

In I-MINDS, the teacher agent interacts mainly with the instructor. The teacher agent helps the instructor deliver instructional content to the students and coordinates the coalition formation process in the classroom. The teacher agent also helps the instructor answer student questions with the *question ranking* and *question grouping* modules.

Question ranking allows the teacher agent to manage Q&A in a large classroom. 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. After scoring, the questions are ranked and displayed to the instructor. The instructor may choose to answer or discard a question. Then depending on the instructor's choice the question ranks are readjusted. Thus, by answering and discarding ranked questions, the instructor implicitly teaches the teacher agent how to better score and rank the questions.

The question grouping module (Khandaker 2004) allows the teacher agent to address multiple similar questions together and profile students who ask similar questions. Two questions are considered similar and grouped together based on the keywords they share and their classification. To classify questions, we use the ApplePie parser (Sekine and Grishman 1995) and the *utterance* classifier program of AutoTutor (Olney et al. 2003, Graesser 1999). When the teacher answers (or discards) a question, the teacher agent automatically answers (or discards) all the similar questions grouped together with that particular question. This functionality helps the instructor address potentially large numbers of similar questions in a large classroom.

Student Agent

In an I-MINDS supported classroom, a student agent serves a unique student. The student agent acts as an interface between a student and the virtual classroom. All the information exchanged between the student and the classroom participants (e.g., questions asked by a student, messages exchanged between peers, etc.) are monitored by the student agent. The student agent also offers surveys to the student to record his/her preferences, and the student agent also exchanges information with the teacher agent and the group agents. With all these collected information, the student agent maintains a dynamic profile of the student to whom it is assigned and a dynamic profile of the peers that the student has interacted with through I-MINDS.

A student is profiled by I-MINDS' agents in several ways. First, student-instructor interactions are profiled mainly based on the quality and classification 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 the messages exchanged among them. This profiling process is carried out by a student agent from the viewpoint of the student it serves. These include the number of messages sent, type of messages, frequency of messages sent, time spent on "idea sketching" and so on. Whenever a student asks a question or performs some kind of collaborative activities (e.g., initiates a forum discussion or participates in sketching out ideas on a digital whiteboard, with both features built into the I-MINDS GUI), his or her profile is updated.

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 (Aronson et al. 1978) which will be discussed later. The group agent monitors the group activities during structured cooperative learning. That includes the number of messages sent among group members, types of messages, self-reported teamwork capabilities, peer-based evaluations as a team member, and evaluation of each team.

Multiagent Coalition Formation

One primary goal of I-MINDS is to support cooperative learning in an environment where students are not able to collaborate face-to-face. Further, our current phase of research and development for our educational application focuses on structured cooperative learning, during which students are guided with specific activities that require collaboration. In particular, the structured cooperative learning that we have incorporated into I-MINDS is the Jigsaw model. Here we first briefly describe what the Jigsaw model is before presenting the multiagent coalition formation algorithm, VALCAM.

Note that our I-MINDS and the VALCAM algorithm are generally designed such that different cooperative learning models can be incorporated. One of our long-term plans is to equip I-MINDS with different cooperative learning models for a teacher to choose from for his or her classrooms.

Jigsaw Learning Model

The learning process in Jigsaw cooperative learning model is as follows. First, assign the students into groups. Second, the instructor divides a problem into different parts (or sections). Third, the instructor assigns a part/section for every student such that members of the same *main* group will have different sections to solve. The students who are responsible for the same section then work together in a *focus* group to come up with solutions to the section to which they have been assigned and develop a strategy for teaching the solutions to their respective group members. Clarke (Clarke 1994) further refined the Jigsaw structure into stages. These stages are (1) *Introduction* of the topic to the class as a whole, (2) *Focused Exploration*: The focus groups explore issues pertinent to the section that they have been assigned, (3) *Reporting and Reshaping*: The students return to their original groups and instruct their teammates based on their findings from the focus groups, and (4) *Integration and Evaluation*: The team connects the various pieces generated by the individual members, addresses new problems posed by the instructor, or evaluates the group product. The goal of Jigsaw is to allow students to learn better by collaborating with each other. The stronger student learns about the solution of the problem by teaching/discussing it with the weaker students. On the other hand, the weaker students learn about the problem solution by method by discussing it with the stronger students. However, not all human students can be expected to work

well with each other. A typical set of learners will have personal preferences for choosing their group members. So, for such a learner group to succeed, the following factors are important: 1) the competence of the group members 2) personal preferences of the group members (about who they would like to form group with). Bearing these factors in mind we have designed the VALCAM algorithm.

VALCAM

To form coalitions (or basic groups for the Jigsaw cooperative learning process), we proposed and designed the Vickrey Auction-Based Learning-Enabled Coalition and Adaptation for Multiagent Systems (VALCAM) algorithm (Khandaker 2005). In VALCAM, we have adapted the auction mechanism to coalition formation for its computationally inexpensive implementation and reduced communication overhead during coalition formation. The multiagent framework—within which our VALCAM algorithm operates—consists of a system agent, a number of user agents and a number of group agents.

The VALCAM process roughly works as follows. The system agent acts as a coordinator and hosts the auction while the user agents, with virtual currency earned, make bids to form coalitions. When a coalition is formed, a group agent is assigned to that coalition. Each group agent monitors the performance and activities of the members of its assigned group. After a coalition has completed its tasks, the group agent evaluates the performance of each user agent as a group member and assigns a group payoff accordingly.

The algorithm also makes use of a virtual currency V in the following manner. The system agent works as the provider and accountant of the virtual currency. Every time the user agents form a coalition and perform the required task, the user agents' performance is evaluated at the individual level and the group level by the system agent and the group agent respectively. After the evaluation, the system agent rewards each user agent's individual performance while each group agent rewards each user agent's performance as a group member.

Here we describe the algorithm. Suppose that A is the set of *user agents*, m is the number of non-overlapping coalitions that will be formed, n is the number of users, and j is the task assigned. The coalition formation algorithm has two parts: VALCAM-S for the system agent and VALCAM-U for each user agent.

VALCAM-S (A, m, n, j)

- 1) **If** this is the first coalition formation among all the users, then (i) Allocate default amount of virtual currency to every user, and (ii) Choose m users randomly from the n users.
- 2) **Else** (i) Allocate default amount of virtual currency to the new users and (ii) Find the topmost m users ranked according to the amount of virtual currency they have and solve any ties randomly.
- 3) Announce the task to the *user agents*

- 4) Randomly assign the selected m users to the m different coalitions
- 5) Announce the opening of the auction to the *user agents* who were *not* chosen in Steps 1 and 2
- 6) **While** there are unassigned *user agents*, perform the following loop:
 - For** $i \leftarrow 1$ to m do
 - (i) Accept bids from the *user agents*, and
 - (ii) **Until** an agent is assigned to the i th coalition do (a) Choose the highest bidder agent, and (b) **If** the chosen agent has enough virtual currency, then subtract the second highest bid from its virtual currency account and assign it to the i th coalition
- 7) **When** the assigned task j is completed do (i) Direct each *group agent* to provide the group payoff $V_g(s, j, t)$ to its student agent members based on their performances in the group, and (ii) Reward each *user agent* with virtual currency $V_s(s, j, t)$ based on the individual performance of the user agent. Then the total payoff for a user agent s for task j at time t is defined as:

$$V_t(s, j, t) = w_{vg} * V_g(s, j, t) + w_{vs} * V_s(s, j, t) \quad (1)$$

where, w_{vg} is the weight associated with group payoff,

$V_s(s, j, t)$ is the total payoff for user agent s for task j at time t for individual performance, w_{vs} and w_{vg} are the weights associated with the individual payoff and group payoff respectively, and $V_g(s, j, t)$ is the total group payoff for user agent s at time t for task j :

$$V_g(s, j, t) = w_{vpc} * V_{pc}(s, j, t) + w_{vac} * V_{ac}(s, j, t) \quad (2)$$

where $V_{pc}(s, j, t)$ and $V_{ac}(s, j, t)$ are the payoffs distributed by the group agent to the user agent based on the potential and actual contribution of the user agent as a member of the group, respectively; w_{vpc} and w_{vac} are the weights associated with contributions. Furthermore,

$$V_{pc}(s, j, t) = \frac{PC(s, j, t)}{\sum_{i \in G} PC(i, j, t)} \quad (3)$$

where $PC(x, j, t)$ is the potential contribution of user agent x at time t for task j and G is the set of all user agents in the group of user s . Moreover,

$$V_{ac}(s, j, t) = \frac{AC(s, j, t)}{\sum_{i \in G} AC(i, j, t)} \quad (4)$$

where $AC(x, j, t)$ is the potential contribution of user agent x at time t for task j and G is the set of all user agents in the group of student s . If a user agent performs well as a group member (e.g., responsive and helpful towards other agents and contributes to team goals.) then this value will be high. This global condition is set to motivate the agents to be a team player.

We also measure the experience between agents s_1 and s_2 , from the viewpoint of s_1 , as:

$$E_r(s_1, s_2, t) = w_{mu} * E_m(s_1, s_2, t) + w_{um} * E_m(s_2, s_1, t) \quad (5)$$

where $E_m(s_1, s_2, t)$ denotes a numeric value computed by s_1 representing its experience in working with s_2 at time t , $E_m(s_2, s_1, t)$ denotes the evaluation score that s_1 receives from s_2 , and w_{mu} and w_{um} are weights. We further compute the average experience measure between two agents s_1 and s_2 over time as:

$$\overline{E}_r(s_1, s_2, t) = \frac{\sum_{t=0}^{q-1} w_t * E_r(s_1, s_2, t)}{T} \quad (6)$$

In Eq. 6, t denotes a time period, T is number of time periods between $t = 0$ and $t = q$, w_t is the weight associated with the time period t .

Algorithm VALCAM-U

- 1) **If** selected as one of the first m users by the *system agent*, **then** Exit
- 2) Post the competence measure C_m to the common bulletin board after a new task announcement, for other prospective group members to read. C_m is basically the self-efficacy of the user agent's ability to perform the announced task. The competence measure of user agent x for task j at time t is defined as:

$$C_m(x, j, t) = \frac{w_{ce} * U_e(x, j, t) + w_{ck} * U_k(x, j, t)}{w_{ce} + w_{ck}} \quad (7)$$

where, $U_e(x, j, t)$ is the mean of previous evaluations received by the user agent x on tasks similar to j at time t , $U_k(x, j, t)$ is the user agent's knowledge/expertise on the assigned task j at time t , and w_{ce} and w_{ck} are weights indicating the relative importance of the two factors.

- 3) **When** acceptance of bid for the i th group is announced by the *system agent*, repeatedly bid an amount of virtual currency $V_b(s, j, t)$ until a bid is won.

Here,

$$V_b(s, j, t) \propto \frac{\sum_{k=1}^m [C_m(k, j, t) + \overline{E}_r(s, k, t)]}{Y} \quad (8)$$

In Eq. 8, $V_b(s, j, t)$ is the virtual currency bid submitted by the student agent s at time t , for task j , $C_m(k, j, t)$ is the competency measure of the k th member of the i th coalition for the task j . Here the user agent s is bidding to join the i th coalition at time t , and the factor Y represents the total number of members in the i th coalition, and \overline{E}_r is from Eq. 5.

Since our agents are self-interested, payoff distribution and virtual currency are important. A user agent's payoff depends on its performance as an individual and its performance as a member of the coalition. Its virtual currency

allows a user agent to bid to join a coalition and to motivate its intentions. A user agent is motivated to bid to join the group that is the most useful to itself—a group that it has worked with before and has received good payoffs as a result of working in the group. If a user agent succeeds in joining the most useful group, then it is likely to be rewarded more than joining a less useful group. This in turn gives the user agent more virtual currency for the next task. To increase the likelihood of joining the most useful group, a user agent’s strategy is to join the group as soon as possible to influence the makeup of the group—discouraging other non-compatible user agents from joining and encouraging other compatible user agents to join.

According to the present learning mechanism (Eq. 8), a user agent would prefer to team up with an agent that it knows is competent from its experience. Specifically, when an agent tries to form a coalition, it considers its past working experience of the coalition member agents, the posted competence of those members and those members’ evaluation of him as a peer.

Thus, using the VALCAM algorithm, an agent learns from its experience in hope of joining a compatible coalition. Note that VALCAM is a soon-enough, good-enough approach and does not guarantee optimality. However, learning from their experiences and payoffs, the agents as a whole improve their coalitions over time.

In the I-MINDS implementation of VALCAM, the *teacher agent* assigned to each instructor assumes the role of the *system agent* and the *student agent* assigned to each student corresponds to the *user agent* during coalition formation.

Our adaptation of VALCAM into I-MINDS was based on the fact that the user agent’s motivation could be related to the student’s motivation. A student wants to learn better and work in a group which consists of students that he/she likes. But joining such a group requires that the user agent assigned to that student has enough virtual currency in its account. So, the student’s motivation to join the best group and learn better is related to the user agent’s motivation to earn as much virtual currency as possible. So, if the student performs well individually and as a group member, the user agent assigned to him/her will earn more virtual currency. This virtual currency balance will mean that the student will be able to join his/her preferred group in the next round.

Implementation

We have implemented I-MINDS in Java (SDK 1.4.2). In particular, we have used Java’s socket functionalities to establish communication among agents, Java’s swing class to create interfaces and Java’s JDBC technologies to connect to our MySQL database. Figure 2 shows the instructor’s interface and Figure 3 shows the student’s interface in I-MINDS.

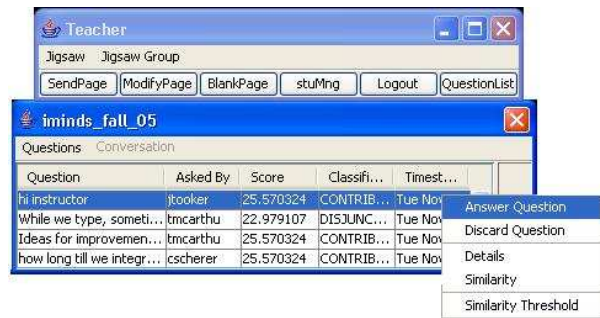


Figure 2: I-MINDS Instructor's Interface

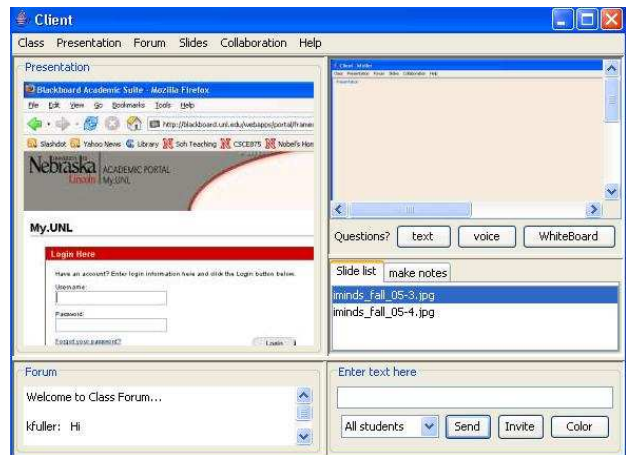


Figure 3: I-MINDS Student Interface

Results

To evaluate I-MINDS, we conducted a pilot study in Spring 2003 to study its feasibility and then deployed I-MINDS in Spring 2005 and Fall 2005 semesters in real classrooms.

The pilot study showed that I-MINDS, as an active (but only primitively intelligent) computer-supported cooperative learning system could be effective in place of face-to-face, traditional instruction. There was evidence that students using I-MINDS were able to achieve a higher learning gain that could be attributed to better questions and discussions supported by I-MINDS.

The course involved was CSCE 155, the first core course of computer science and computer engineering majors (i.e., CS1). The course has 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 programming problems. These were introductory problems about Debugging, Inheritance, UML and Recursion. For example, in the UML session, the students were required to design a resource management system from the given requirements. This main task was then divided into two sub tasks: designing class diagrams and designing use cases. For each semester, there were 2-3 lab sections where each section had about 15-25 students.

Our study utilized a control-treatment protocol. 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 (tasks divided into subtasks by the instructor). We conducted the study for several lab sessions, covering topics in debugging and testing, inheritance and polymorphism, Unified Modeling Language (UML), and recursion.

The students of both the control and treatment sections were required to complete the tasks and subtasks in the four Jigsaw phases as discussed earlier. 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 group formation (coalition formation) using the VALCAM algorithm. 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. Then the three Jigsaw Phases were carried out in order. During these three phases, the student agents and the group agents monitored and guided 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 graded by the instructor and 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 (Soh et al. 2005).

Table 2 shows the post-test scores of the control (i.e., Jigsaw without I-MINDS) and treatment (i.e., Jigsaw with I-MINDS) sections for both semesters.

The results indicate that students using I-MINDS for the Jigsaw activities were able to obtain comparable 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 focused exploration and reporting phases 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 collaboration. Furthermore, on average, 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 speculate 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. On the other hand, we also observe that students in the treatment sections seemed to improve over time, and their performance seemed to eventually overtake that of the control sections’ over time—indicating that VALCAM, due to its learning mechanism, might have been effective in forming better and better coalitions over time. These promising results will require more data collection in the near future to attain higher significance.

Table 2: Student post-test scores for the control (Jigsaw without I-MINDS) and treatment (Jigsaw with I-MINDS) sections

Spring 2005				
Session	Control Section		Treatment Section	
	Mean	Stdev.	Mean	Stdev.
1	7.06	1.83	6.10	1.79
2	5.00	2.41	7.63	1.72
3	8.83	2.85	9.00	1.50
Mean	6.96	2.36	7.57	1.67
Fall 2005				
Session	Control Section		Treatment Section	
	Mean	Stdev.	Mean	Stdev.
1	7.33	1.28	7.71	1.68
2	8.12	1.50	7.81	0.95
3	8.14	0.86	8.53	0.87
4	8.81	1.37	9.14	0.94
Mean	8.10	1.25	8.29	1.11

Next we look at the results of the Self-Efficacy Questionnaire (SEQ) survey. The SEQ survey was conducted among the students after the introduction phase. Students enter their competency of completing a particular task. This contributes to Eq. 7 in the VALCAM-U algorithm. Table 3 shows the results.

We observe that for both semesters, students in the treatment sections were on average less confident than the students in the control section about their ability to solve the assigned task before the Jigsaw cooperation started. This is interesting. Table 2, as discussed earlier, shows that the students in the treatment sections performed comparably and eventually overtook those students in the control sections in terms of their post-test scores, even though Table 3 shows that these were the same students who had

lower self-efficacy. This further lends credence to I-MINDS as an intelligent CSCL system.

Table 3: Self-Efficacy Questionnaire Results

Spring 2005				
Session	Control Section		Treatment Section	
	Mean	Stdev.	Mean	Stdev.
1	33.33	3.97	32.36	4.00
2	35.00	3.30	33.63	5.14
3	34.83	2.28	33.50	3.96
Mean	34.39	3.18	33.16	4.37
Fall 2005				
Session	Control Section		Treatment Section	
	Mean	Stdev.	Mean	Stdev.
1	31.72	4.26	23.64	5.55
2	32.50	6.48	32.00	3.63
3	34.35	6.54	35.89	3.89
4	35.56	6.55	32.40	3.60
Mean	33.53	5.95	30.98	4.16

Next, we look at the surveys in terms of peer rating. The Peer Rating Questionnaire (PRQ) surveys were conducted in both control and treatment sections *after* each lab session was completed. The PRQ is designed to quantify the compatibility of the group members after they have gone through the cooperative learning process. We then averaged the average peer rating scores that each student gave to his or her group members. This average for each session can be used as a measurement of how well the team members in each section were able to work with each other. Table 4 shows the results of the PRQ surveys.

Table 4: Peer Rating Questionnaire Results

Fall 2005				
Session	Control Section		Treatment Section	
	Mean	Stdev.	Mean	Stdev.
1	42.10	2.73	32.45	5.78
2	36.62	7.05	37.72	4.60
3	39.91	4.80	34.63	8.08
Mean	39.54	4.86	34.93	6.15
Spring 2005				
Session	Control Section		Treatment Section	
	Mean	Stdev.	Mean	Stdev.
1	35.39	2.30	33.71	4.69
2	34.87	5.32	35.80	12.21
3	36.03	3.19	36.37	5.18
4	37.53	3.37	37.25	3.62
Mean	35.95	3.54	35.78	6.42

As evidenced in the scores, students in the control section rated their peers better (higher means) and more consistently (lower standard deviation values) 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 GUI features and multimedia capabilities to fully capture real-time characteristics of interactions.

On the other hand, we see indications that students in the treatment section for the Spring 2005 section seemed to rate their peers better over time (from 33.71 to 35.80 to 36.37 and 37.25) and seemed to rate their peers more consistently as well. This might be due to the ability of the coalition formation algorithm in forming better groups over time. Given sufficient amount of time the VALCAM algorithm for coalition formation would allow the users to form groups with their favorite peers. So, over time the users were able to interact with others and evaluate each others as team members. This evaluation in the form of PRQ then helped them choose better team members in the future sessions.

Next we discuss the Team-Based Efficacy Questionnaire (TEQ) surveys, which were collected after each lab based on a set of questions designed to measure how a student viewed how well its group had performed. Table 5 shows the results of the TEQ conducted after every session.

Table 5: Team-Based Efficacy Questionnaire Results

Spring 2005				
Session	Control Section		Treatment Section	
	Mean	Stdev.	Mean	Stdev.
1	31.80	2.58	27.72	5.08
2	30.87	3.38	29.18	2.63
3	30.08	3.02	28.25	4.02
Mean	30.92	2.99	28.38	3.91
Fall 2005				
Session	Control Section		Treatment Section	
	Mean	Stdev.	Mean	Stdev.
1	27.22	4.37	23.64	5.55
2	26.75	6.66	25.87	8.33
3	29.14	5.47	25.76	5.43
4	29.12	4.52	26.78	8.15
Mean	28.05	5.25	25.51	6.86

We observe that students in the control section approved of their team-based activities more than the students in the treatment section. There are two possible explanations. First, the ease of face-to-face interactions gave the impression that the group was doing better, which is consistent with our earlier observation with the peer rating results. Second, how the student agents form their coalitions did not necessarily meet the students' preference. Note that a student did not have access to other survey results, including how his or her group members thought of his or her as a peer. However, the student agent did and perused this information in its bidding for the most useful/compatible group. Further studies will be necessary to investigate how a student's perception of a group correlate with or influence the actual quality of a group.

Conclusions

We have described I-MINDS, a system that uses multiagent intelligence and coalition formation to provide computer-supported cooperative learning in education. 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. We have also described the VALCAM algorithm for forming coalitions. To demonstrate the effectiveness of I-MINDS, we have presented the results of incorporating Jigsaw Framework in I-MINDS in real-time classroom situations.

The deployment results in our CS1 lab for two semesters show that, although face-to-face activities were perceived to be better by the students, I-MINDS-supported activities were able to produce better or comparable performance (in terms of post-test scores) and were able to yield generally more tightly clustered performance among the students. We also have indications that the VALCAM algorithm was able to learn to form better groups over time in terms of peer rating and team-based efficacy.

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.

Acknowledgements

The authors would like to thank the National Center for Information Technology in Education (NCITE) and NSF (SBIR grant DMI-044129) for the research funding. The authors would also like to thank X. Liu, P. Vemuri, S. Namala, and X. Zhang for their programming work.

References

Aronson, E., Blaney, N., Sikes, J., Stephan C., and Snapp, M. eds. 1978. *The Jigsaw Classroom*. Beverly Hills, CA: Sage.

Betbeder, M.-L., Tchounikine, P. and Laënnec, A. 2003. Symba: A Framework to Support Collective Activities in an Educational Context. In *Proc. ICCE*, 190-196. Hong Kong, China.

Caverly, D. C. and L. MacDonald (2002). Online Learning Communities. *J. Developmental Education* **25**(3):36-37.

Chan, S. C.-F., Leung, C. W.-K., and Ng, V. T.-Y. 2003. GroupUML: A PDA-Based Graphical Editor to Support

Real-Time Collaboration in Student Group Projects. In *Proc. ICCE'03*, 221-223, Hong Kong, China.

Chang, C.-C. 2003. Implications and Issues of Building a Distributed Web-Based Learning Community. In *Proc. ICCE'03*, 224-233, Hong Kong, China.

Clarke, J. eds. 1994. Pieces of the Puzzle: The Jigsaw Method. In *Handbook of Cooperative Learning Methods*. Westport, CT: Greenwood Press.

Graesser, A. C., Wiemer-Hastings, K., Wiemer-Hastings, P., Kreuz, R. and Tutoring Research Group. 1999. Autotutor: A simulation of a human tutor. *J. Cognitive Systems Research* 1:35-51.

Khandaker, N. 2005. VALCAM – An Auction Based Learning Enabled Multiagent Coalition Formation Algorithm for Real-World Applications. M.S. Thesis, CSE, Univ. Nebraska, Lincoln, NE.

Khandaker, N. 2004. Question Classification and Question Similarity in I-MINDS, Independent Research Report, Dept. Computer Science & Engineering, Univ. Nebraska, Lincoln, NE.

Liu, X., Zhang, X., Al-Jaroodi, J., Vemuri, P., Jiang, H. and Soh, L.-K. 2003a. I-MINDS: An Application of Multiagent System Intelligence to On-Line Education. In *Proc. IEEE-SMC*, 4864-4871. Washington D.C.: IEEE.

Liu, X., Zhang, X., Soh, L.-K., Al-Jaroodi, J. and Jiang, H. 2003b. A Distributed, Multiagent Infrastructure for Real-Time, Virtual Classrooms. In *Proc. ICCE*, 640-647. Hong Kong, China.

Olney, A., Louwerse, M., Mathews, E., Marineau, J., Hite-Mitchell, H and Graesser, A. 1995. Utterance classification in Autotutor. In *Proc. HLT-NAACL Workshop*, 1-8. Philadelphia: Association for Computational Linguistics.

Salcedo, R. M., Yano, Y., Miyoshi, Y., and Ogata, H. 2003. Collaborative Spaces in a Distributed Digital Library. In *Proc. ICCE'03*, 136-129, Hong Kong, China.

Sekine, S. and Grisman, R. 1995. A Corpus-Based Probabilistic Grammar with Only Two Non-Terminals. In *Proc. 4th International Workshop on Parsing Technologies*, 216-223. Prague, Czechoslovakia.

Soh, L.-K., Jiang, H. and Ansoorge, C. 2004. Agent-Based Cooperative Learning: A Proof-of-Concept Experiment. In *Proc. SIGCSE*, 368-372. Norfolk, Virginia.

Soh, L.-K., Khandaker, N., Liu, X. and Jiang, H. 2005. Computer-Supported structured cooperative learning. In *Proc. ICCE*, 420-427. Singapore.

Soh, L.-K., Khandaker, N., Liu, X., and Jiang, H. 2006. A Computer-Supported Cooperative Learning System with Multiagent Intelligence, accepted to *AAMAS'2006*.

Sridharan, B., Kinshuk, H., and Hong, H. 2003. Agent Based System to Capture, Discover and Retrieve Knowledge from Synchronous and Asynchronous Modes of Learning, In *Proc. ICCE'03*, 759-760, Hong Kong, China.