

Coaching Collaboration by Comparing Solutions and Tracking Participation

María de los Angeles Constantino-González

Center for Artificial Intelligence
Monterrey Institute of Technology (ITESM)
E. Garza Sada 2501 Sur
Monterrey, N.L. 64849, México
aconstan@cia.mty.itesm.mx

Daniel D. Suthers

Department of Information and Computer Sciences
University of Hawai'i at Manoa
1680 East West Road, POST 303A
Honolulu, HI 96822, USA
suthers@hawaii.edu

Abstract: This paper reports a new approach to coaching collaboration in a synchronous distance learning context. Prior work on supporting collaboration has relied largely on comparing student discourse to models of collaborative discourse. Comparison of student work to expert solutions is prevalent in individual coaching paradigms. Although these approaches are valuable, our approach evaluates the potential contribution of tracking student participation and comparing students' individual and group solutions. Our theoretical motivation is that conflicts between individual and group solutions constitute learning opportunities, provided that students recognize and address these conflicts. The coach encourages such negotiation when differences are detected, and also encourages participation in other ways. Our evaluation relied primarily on expert judgement and secondarily on student reactions to the coach. Results show that the quality of the generated advice was good; however, other knowledge sources should be consulted to improve coverage of advice to a broader range of situations and advice types.

Keywords: collaborative distance learning, intelligent agents, entity-relationship modeling, socio-cognitive conflict theory

1. Introduction

Collaboration is integral to today's organizations, which require individuals who can work together to solve complex problems and share their own knowledge and experiences with others. Collaborative skills can be learned, and it is therefore essential to provide individuals with appropriate learning opportunities. Support for collaboration is especially important in distance learning, because the lack of face-to-face interaction complicates the collaboration. Although remote students sometimes work in groups, there is little evaluation of the collaboration process and the students' collaborative skills. Abrami (1996) states "...with social and intellectual isolation, students may fail to develop and refine those cognitive and interpersonal skills increasingly necessary for business and professional careers."

Collaborative learning studies in the classroom show that properly designed collaborative learning techniques help students to improve their achievement and develop their critical thinking and cooperative behavior (Slavin, 1995; Johnson & Johnson, 1994). Nevertheless, it is not sufficient to provide distance learners with a communication channel. Facilitators should monitor students' collaboration in order to guide participants in the application of collaborative skills. However, it is hard for a facilitator to support collaboration when many teams have to be monitored. An intelligent system could be helpful.

Several systems have been designed to encourage participation and facilitate group discussion with intelligent support, such as C-CHENE (Baker & Lund, 1996), McManus & Aiken's (1995) Group Leader Tutor, IDLC's Expert System Coordinator (Okamoto *et al.*, 1995), and BetterBlether (Robertson *et al.*, 1998). All of these systems use restricted menu-driven or sentence-opener interfaces in order to understand students' interaction, and give guidance based on an ideal model of dialogue. Dialogue-based support provides several advantages, such as potential applicability to any subject matter area, automated interpretation of students' interactions, and restriction of discussion moves and learning interactions to those believed to be productive for learning. However, these systems present some disadvantages, such as restricting the type of communicative acts, slowing the communication process, and misinterpreting students' dialogue when students use the interface buttons incorrectly. It would be advantageous to increase the repertoire of ways to provide automated support.

Our work seeks to facilitate effective collaborative learning interactions, particularly with respect to the recognition and resolution of conflicts between students' problem solutions, with minimal reliance on restricted communication devices such as sentence openers. In this paper, we evaluate the feasibility of gen-

erating advice based primarily on comparing students' individual and group solutions and on tracking student participation (contributions to the group diagram). In this way, our work differs from previous work in this area, which has supported interaction by analyzing collaborative dialogue. Our approach is similar to the one proposed by Mühlenbrock & Hoppe (1999) in which multiple users are monitored and an analysis of interaction moves within a shared window is done. However, our approach also considers students' individual work to identify conflicts, based on CSCL learning theories. Other previous studies have used automated coaches to give advice when a student's solution differs from an expert's solution. In contrast, our work evaluates the possibility of giving advice without comparing student work with an expert solution.

Our theoretical motivation is based on pedagogical theories that explain how social interaction mediates learning. According to the Socio-Cognitive Conflict Theory, students learn from disagreements when they identify and resolve conflicts in their viewpoints, present alternatives, and request and give explanations. Cognitive Dissonance Theory states that the existence of disagreement among members of a group produces cognitive dissonance in the individual, who experiences pressure to reduce this dissonance, leading the individual to a process of social communication and revision of his position. The value of the disagreement depends less on the correctness of the opposing position than on the attention, thought processes and learning activities it induces. A coach monitors participation and detects differences between students' individual and group work and encourages learning interactions when differences are detected, or when other situations warrant certain advice.

Entity-Relationship (ER) modeling, one of the most critical phases in the development of information systems, was selected as an appropriate task for this research for several reasons. Database design is a collaborative activity: designers and database users collaborate to produce a conceptual schema that meets the information needs of an organization (Batini, Ceri & Navathe, 1992). Different solutions are possible due to different assumptions or misconceptions. Therefore students may have genuine differences to discuss. Research has found that ER Modeling is a complex task for novices (Batini, Ceri & Navathe, 1992; Gordon & Hall, 1998). A good proportion of novice errors is due to students' acceptance of the initial solution without considering alternatives (Batra & Antony, 1994).

We designed a computer-mediated environment in which students construct Entity-Relationship diagrams as solutions to database modeling problems. Students begin by constructing their entity-relationship diagrams alone. Later they work in small groups to agree upon a group solution. A software coach identifies differences between students' solutions based on a minimal understanding of the ER and application domains. The coach tries to lead learners into particular kinds of interactions expected to lead to learning. The remainder of this paper describes the web-based collaborative learning environment, the computer coach, and our evaluation of its adequacy and of the role of knowledge in coaching collaboration.

2. COLER

COLER is a Web-based collaborative learning environment in which students can solve database-modeling problems while working synchronously in small groups at a distance.

2.1 COLER's Interface

COLER provides four different modes of operation according to the type of user (student or professor) and the selected type of session (individual or group). COLER's student group interface is shown in Figure 1. The *problem description window* (upper center) presents an entity-relationship modeling problem. Students construct their individual solutions in the *private workspace* (upper right). They use the *shared workspace* (lower center) to collaboratively construct ER diagrams while communicating largely via the *chat window* (lower right). They can use a **HELP button** (upper left) to get information about Entity-Relationship Modeling. A *team panel* (middle left) shows which teammates are already connected. Only one student, the one who has the pencil, can update the shared workspace at a given time. The *floor control panel* (bottom left) provides two buttons to control this workspace: **ASK/TAKE PENCIL** and **LEAVE PENCIL**. Additionally, this panel shows the name of the student who has the control of this area and the students waiting for a turn. An *opinion panel* (middle right) shows teammates' opinions on a current issue. This area contains three buttons: **OK**: Total Agreement, **NOT**: Total or Partial Disagreement, and **?**: Not sure, Uncertainty. When a button is selected, students have the option of annotating their selection with a justification. Opinion button selections are displayed in the chat area (along with any optional justifications) in order to correlate these opinion-expressing actions with the chronology of the chat discourse. Opinion button selections are also displayed in the opinion panel to provide students with a persistent summary of their teammates' current

opinions. A *personal coach* (upper left) gives advice in the chat area based on students' participation and group diagram construction. Although several suggestions may be computed at a certain time, only one is shown in the chat area. The others may be obtained by pressing the **SUGGESTIONS** button, which is disabled if the coach does not have any advice to offer.

COLER is designed for sessions in which students first solve problems individually and then join into small groups to develop group solutions. The initial problem solving helps ensure individual participation and provides differences between students' solutions that form the basis for discussion. The private workspace also enables students to try solutions without feeling they are being watched. When all of the students have indicated readiness to work in the group, the shared workspace is activated, and they can begin to place components of their solutions in the workspace. This may be done either with **COPY/PASTE** from private workspaces or by making new structures in the shared workspace. After each change to the workspace, the changed object is highlighted in yellow; then, students are required to express their opinions using the **OK/NOT/?** buttons before making subsequent use of the shared workspace.

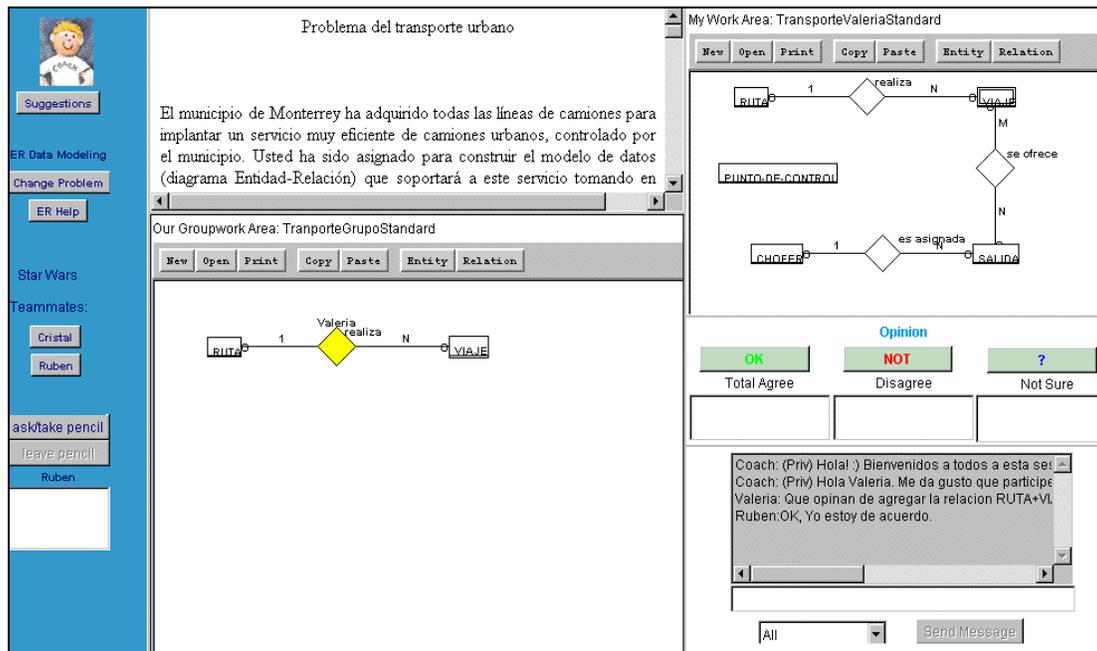


Figure 1: COLER Collaborative Student Interface

COLER is based on the open architecture for collaborative learning systems designed by Suthers & Jones (1997) and originally used for the implementation of the Belvedere software for collaborative critical inquiry (Suthers, et al., 1997). Details of architecture and formative design are given in Constantino-González and Suthers (2000) and Constantino-González (2000).

3. COLER's Coach

COLER's personal coach is a pedagogical agent that encourages students to discuss and participate during collaborative problem solving. The coach's student is called MyCoachedStudent (MCS). COLER constantly observes the actions of MCS in the learning environment. It also observes participation in the shared workspace and in chat discussions (no natural language interpretation is attempted), and selections of opinion buttons. Using this information, COLER decides whether to give advice.

3.1 Personal Coach's Knowledge

COLER requires minimal domain and problem-specific knowledge for the detection of learning opportunities. Its *domain knowledge* includes heuristic knowledge about significant differences in ER diagrams and the procedures for identifying these differences. We also defined some problems with the quality of ER diagrams and the procedures to recognize them. *Problem-specific knowledge* consists of a glossary for the

problem that students will solve. The glossary is used to match diagrams and identify differences, compensating for the coach's lack of natural language understanding capability.

The *knowledge for coaching collaboration* consists of the ability to recognize relevant learning opportunities and to provide advice that encourages students to take these opportunities. Advice types and categories were defined based on the collaborative learning literature and formative studies. Seven advice *categories* are defined in the present version of COLER (Table 1). The first two categories, *Discussion* (in chat) and *Participation* (in the group workspace), are the main categories related to coaching collaboration. *Feedback* messages are related to student's pressing of COLER opinion buttons. The *ER Modeling* category includes suggestions related to some common errors in the domain. The *Self-Reflection* category consists of suggestions that individuals think about a problem or situation. Besides using advice from these categories, COLER can use messages for *welcoming* and *saying goodbye*.

Several *types* of advice were defined and classified according to each of these categories (Table 1). For each advice type, several advice *templates* were defined using different wording to provide linguistic variety. They can be contextualized by binding variables from the current situation, including the student's name, the object type (e.g. entity, relationship), the object's name, and the problem type (e.g. disconnected entity, no key defined). An example template (translated from the Spanish) follows:

\$MyStudentName, \$ObjectName \$ObjectType proposed in the diagram is different from what you've got. If you do not agree with this, you should express and justify your viewpoint.

Several heuristic control strategies were specified to define COLER's general behavior concerning when and what advice to give. *Category Preferences* are used to select between alternate advice. These preferences are prioritized differently according to the group dynamics. *Discussion Intensity*, *Participation Balance*, *Time on Task*, and *Waiting for Feedback* strategies are all computed based on a set of parameters. Details are provided in Constantino-González (2000).

Table 1: Coach Advice Types

<i>Category</i>	<i>Abbreviation</i>	<i>Advice Type Description</i>
Discussion	ED	Express Disagreement
	AE	Ask for Explanation
	AJ	Ask for Justification
	GE	Give Explanation
	GJ	Give Justification
	EU	Express Uncertainty
	AA	Analyzing Alternatives
	RA	Reflect with teammates about...
Participation	GC	General Contribution
	SC	Specific Contribution
	CT	Continue working on Task
	GP	Explain, in general, the importance of participation
	LO	Listen to Others
	LP	Let Others Participate
	IP	Invite others to Participate
	LM	Listen to Others, Mandatory
Feedback	LC	Ask a teammate to let you contribute
	AF	Ask for Feedback
Self-Reflection	GF	Give Feedback
	CD	Check Own Discrepancies
ER Modeling	ER	Entity-Relationship Modeling: Connect a disconnected Entity, draw a relationship, add an entity or attribute, define a key.
	RW	Review Work Completeness
Welcome	IW	Individual Welcome
	GW	Group Welcome
Goodbye	IG	Individual Goodbye
	GG	Group Goodbye

3.2 Personal Coach's Architecture

The main module, the *Collaboration Supervisor* consists of two modules: the Advice Generator and the Advice Selector. The *Advice Generator* computes the set of appropriate advice based on AND/OR situation trees. Several pieces of advice might be suggested for any given event. The *Advice Selector* chooses the most appropriate advice from this set based on prioritized preferences and random choice. These modules rely on three helper modules. The *Differences Recognizer* identifies opportunities for students to collaborate by recognizing semantically significant differences between individual and group ER diagrams. It uses sub-graph matching to either find differences specifically related to the currently added object or find all “extra work” that the student can contribute to the group. The *Participation Monitor* attends to the activity in the group diagram. It attends to time-triggered events, such as inactivity in the group area and MCS having the control of the group area for a long time. Group diagram events, such as object addition to the group diagram, are also monitored to determine whether each student is participating too much or too little. The *Diagram Analyzer* is a simple module that identifies participation opportunities based on the detection of problems in the quality of the ER group diagram. It uses syntactic and semantic information. A detailed description of the architecture modules is given in Constantino-González (2000).

3.3 Example Scenario

An example of the coach's operation is presented below. Figure 3 shows the diagram constructed by the team formed by David, George and Frank. Figure 4 illustrates George's diagram. Frank, George's teammate, has just added the Project+Researcher relationship (participates) to the group diagram. Then, George pressed OK button to indicate his agreement.

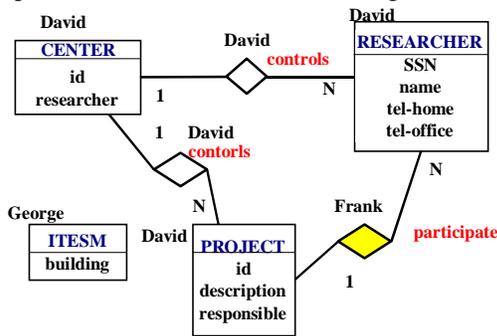


Figure 3: Group Diagram

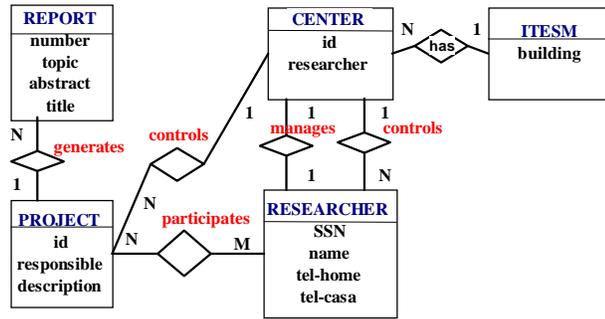


Figure 4: George's Diagram

The *Participation Monitor* concludes that George, who has one contribution, has not participated enough. Then, the appropriate branches of the AND/OR situation tree are analyzed and the following advice types are generated: GC, GP; SC(REPORT), SC(CENTER+ITESM:has), SC(CENTER+RESEARCHER:manages); ER(ITESM disconnected), ER(CENTER,key). From the SC types one is selected randomly from the more appropriate ones (e.g. the two relationships: CENTER+ITESM, CENTER+RESEARCHER). The *Difference Recognizer* compares George's and the group's diagrams, finding that George defined PROJECT+RESEARCHER relationship with cardinality N to M instead of 1 to N, and that he labeled the relationship with *participates* instead of *participate*. Considering that the feedback given was OK and there was no extra evidence of disagreement, additional advice types generated by the AND/OR situation tree are AE, AA, EU and RA.

Advice Selection eliminates the type of the advice most recently given (AA in the present example), and then randomly selects one advice type from each leaf of the AND/OR tree. In our example, the result is as follows: GP; SC(Center+ITESM:has); ER(Center, key) and RA. Examples of linguistic realizations of the advice (advice templates with variables bound) are shown below:

GP: *George*, participation is a learning opportunity. I suggest you leverage it. Come on, participate!

SC: *George*, you could share your work with your teammates by adding *CENTER+ITESM relationship* to the diagram

ER: *George*, you could *define the key* of the *CENTER* entity.

RA: *George*, *PROJECT+RESEARCHER* relationship has been just added by *Frank*. What do you think about it? Is it correct? I suggest you discuss it with your teammates.

Before this list is sorted according to the preferences, the group's participation is evaluated to adjust the preference priorities. Since there is a problem in participation, the priority of a Participation preference is increased. The following order is produced: SC, RA, ER, GP. Therefore the coach then gives the SC advice shown above. The rest of the advice patterns generated are stored for future use and made available for advice on demand.

4. Evaluation

COLER had undergone various forms of evaluation. The evaluations reported here address expert judgment of advice quality, students' reactions to advice, and contributions of knowledge sources. Future publications will report evaluation of advice generation and selection algorithms and the relationship between group functioning and COLER's advice.

4.1 Summary of Method and Procedure

This laboratory evaluation of COLER involved participants who had the right level of domain knowledge for using the system: ITESM students taking a database course. Our domain expert, a computer science professor, was also present in two sessions. Five sessions were conducted to generate data and scenarios for the different types of evaluations. In each of these sessions, three students were presented with a simple database design problem. They first solved the problem individually, and then convened to construct a group solution. Software instrumentation recorded all of the activities of the students and of COLER's coach. The pilot study and the two sessions in which the Expert was present were used for preliminary evaluation, detecting some problems in COLER's user interface and coach algorithms. The last three sessions, in which the expert was not present, were used to evaluate COLER's algorithms and the quality of its advice. These sessions involved a total of 72 advice events. Of these, 34 were Participation, 23 Discussion, 6 Self-Reflection, and 9 Feedback advice.

4.2 Expert Judgement Results

Documents were generated to describe the chronological sequence of events of the collaborative session in reference to a specific student, and the current state of the environment associated with each event. The expert evaluated each of the three sessions that he did not witness by analyzing the documents generated for each student. These documents provided the expert with precisely the same information that was available to COLER's coach. For example, the documents included the existence of chat contributions but not their contents. The expert first indicated the advice he would give in each situation. He then ranked the advice *generated* by COLER and indicated whether this advice was "reasonable," "so-so," or "not worth saying." He was not told the actual advice *selected* by COLER until his judgments were complete.

Of the advice actually selected by COLER, the expert judged 71% as "reasonable" and 16% as "so-so." Advice was judged as "so-so" primarily for reasons of inappropriate wording used in COLER's advice patterns. Reviewing the wording with the expert could solve this problem. The non-reasonable advice (13%) was attributed to two main problems: changes in the environment that made the advice inapplicable, and spurious mismatches due to spelling errors. Defining the conditions for each specific advice type that should be reviewed before giving the advice could solve the first problem. Spelling errors could be managed by devaluing the importance of differences in relationship's names for generating "Check discrepancy" advice, or by using a distance metric between the spellings.

Turning to coverage, 67% of the advice given by the expert was not given by the coach. Of this advice, 69% would require new advice types and new branches in the AND/OR situation tree, 21% involved situations already defined in the AND/OR tree but requiring new advice types attached to them, and 10% involved advice that COLER could give with minor adjustments to parameters. A new category of advice, "Social Interaction," is needed to establish a closer relationship between COLER coach and the student. This category could include different advice types such as thanking the student for listening to advice, and otherwise commenting on student actions. Some existing advice types need to be extended to mention a specific context, such as suggesting that students reflect on a specific difference or inviting someone in particular to participate. The findings also suggested situations in which a new "Self-Reflection" advice type could be given.

4.3 Student Reactions

An analysis of the chat transcripts and videotapes provided information about the effects of COLER's advice on students' behavior. Findings indicated that the students took 40% of the total advice instances; 28% were applicable but ignored; 21% were no longer needed due to changes in the situation; and student response to 11% of the advice could not be determined.

Questionnaires administered to the 13 students in all 5 sessions indicated that students found several types of advice to be useful, while several other types of advice were found sometimes useful and sometimes irrelevant, such as the "Continue Working on Task" and the "Review Work" suggestions. Almost all students believed that advice was given at appropriate times since it occurred during or immediately after the events. However, some students indicated that some advice was given just after the action suggested was performed, and sometimes the advice interrupted chat continuity. Concerning advice frequency, 69% of students think it was appropriate, 23% think it was low and 8% think it was high. Most of the students thought that the presence of a coach during the session helped guide and coordinate the collaborative session and establish the group dynamics required in collaborative learning. Most students said they reaffirmed their ER knowledge and learned about collaboration during the session. Students suggested additional types of messages and indicated that sometimes advice should be given to the whole group instead of just to individuals.

4.4 Roles of Knowledge Sources

The contribution of knowledge sources to generation of advice judged to be "reasonable" ranks as follows: Feedback Tracking and Feedback timeout (49%), Participation Balance (48%), Significant Differences and Glossary (41%), Time on Task (40%), Chat Tracking (37%), Discussion Intensity Parameters (29%) Category and Sort Preferences (22%), Pencil Tracking (14%) and Common Problems in ER diagram (2%). Some knowledge sources were used to generate multiple categories of advice (hence the percentages reported above sum to greater than 100) while others were used only to generate a specific advice category. The generation of reasonable advice (mostly Discussion and Participation) in this study required the conjunction of several types of knowledge (e.g. Significant Differences and Glossary, Participation Balance) and confirmed our hypothesis that monitoring of problem solving activity could be used to generate collaboration advice.

We evaluated the potential role of natural language understanding and expert domain knowledge by having the expert read the students' chat transcripts at the end of the session and review his suggestions for a second time. (Chat contents were not available during his initial assessments.) He indicated that there wouldn't be significant changes in what he said as a coach because he and COLER emphasize collaboration and participation, because ER is a complete modeling technique to represent the design, and because the problem type considered in these sessions clearly specified the user requirements. If the problems had been less well specified, natural language understanding may have been necessary in order to track the substantial discussion required to negotiate agreement.

5. Summary

This work is part of a research agenda that seeks to characterize the knowledge needed to facilitate collaborative learning processes. The work follows an incremental research strategy by implementing and evaluating a small number of simple knowledge sources to understand their value before incorporating additional or more complex knowledge sources. We focused on how much leverage can be gained by a basic ability to detect semantically interesting differences between two representations of problem solutions, coupled with simple tracking of individual's quantity of participation and discussion. We showed that reasonable collaboration advice can be generated without the need for expert solutions or discourse understanding. (The addition of these knowledge sources would improve the quality and range of advice generated and selected by the system, at the cost of considerable additional knowledge engineering.) This approach should generalize to all domains in which students construct formal representations of problem solutions that can be compared for significant differences. We encourage others to test the value of the approach in other domains that have this property.

6. Acknowledgments

We thank our colleagues at the Learning Research and Development Center of the University of Pittsburgh for their support; in particular, Alan Lesgold for hosting the first author as a visiting scholar and the second as his research associate, Sandy Katz for discussions about collaborative learning, and Kim Harrigal and Dan Jones for their assistance with the implementations. We also thank Jose G. Escamilla of ITESM for his advisorship of the first author, Jose I. Icaza of ITESM for acting as our domain expert, Moraima Campbell of ITESM for her support in user interface design, and visiting students Vincent Trifot and Justin Peltier for implementing part of the personal coach and the Professor module. The Center for Artificial Intelligence of ITESM provided research facilities. During this research, Constantino-González was funded by ITESM Campus Laguna and by CONACYT, and Suthers was funded by the Presidential Technology Initiative (while at LRDC) and by NSF's Learning and Intelligent Systems (while at the University of Hawai'i).

7. References

- Abrami, P.C. and Bures, E.M. (1996) Computer Supported Collaborative Learning and Distance Education, Reviews of lead article. *The American Journal of Distance Education* 10(2): 37-42.
- Baker, M.J. and Lund, K. (1996, October). Flexibly Structuring the Interaction in a CSCL environment. In P. Brna, A. Paiva & J. Self (Eds.) *Euro-AIED '96 Conference Proceedings (European Conference on Artificial Intelligence and Education)*, Lisbon.
- Batini, C., Ceri, S. and Navathe, S. B. (1992). *Conceptual Database Design: An Entity-Relationship Approach*. Benjamin/Cummings, Redwood City, California.
- Batra, D. and Antony, S. R. (1994). Novice Errors in Conceptual Database Design. *European Journal of Information Systems*, Vol. 3, No. 1, pp. 57-69.
- Constantino-González, M. A. (2000). *A Computer Coach to Support Synchronous Computer-Mediated Collaborative Learning*. Unpublished dissertation, ITESM (Instituto Tecnológico y de Estudios Superiores de Monterrey), México.
- Constantino-González, M.A. and Suthers, D.D. (2000). A Coached Collaborative Learning Environment for Entity-Relationship Modeling. In Gauthier, G., Frasson, C. and VanLehn, K. (Eds.) *Intelligent Tutoring Systems: Proc. of the 5th International Conference, ITS 2000*, Montreal, June, pp 324-333.
- Gordon, A. and Hall, L. (1998). A Collaborative Learning Environment for Data Modeling. In *Proc 1998 Florida Artificial Intelligence Research Symposium*. Sanibel Island, FL: pp. 158-62. AAAI Press.
- Johnson, D.W. and Johnson, R.T. (1994). *Learning Together and Alone*, Englewood Cliffs, NJ: Prentice Hall.
- McManus, M.M. and Aiken, R.M. (1995). Monitoring Computer Based Collaborative Problem Solving, *International Journal of Artificial Intelligence in Education*, 6(4): 308-336.
- Muehlenbrock, M. and Hoppe, U. (1999). Computer Supported Interaction Analysis of Group Problem Solving. In *Proc. of the Computer Support for Collaborative Learning (CSCL) 1999 Conference*, C. Hoadley & J. Roschelle (Eds.) Dec. 12-15, Stanford University, Palo Alto, California. Mahwah, NJ: Lawrence Erlbaum Associates.
- Okamoto, T., Inaba, A. & Hasaba, Y. (1995). The Intelligent Learning Support System on the Distributed Cooperative Environment. In J. Greer (Ed.), *Proc. of AI-ED 95 - 7th World Conference on Artificial Intelligence in Education* (pp. 210-218). Charlottesville: Association for the Advancement of Computing in Education (AACE).
- Robertson, J., Good, J. & Pain, H. (1998). BetterBlether: A Computer Based Educational Communication Tool, *International Journal of Artificial Intelligence in Education*. 9(3-4): 219-236.
- Slavin, R. E. (1995). *Cooperative Learning*. Allyn and Bacon, 2nd. edition.
- Suthers, D.D. and Jones, D. (1997, August). An Architecture for Intelligent Collaborative Educational Systems. In B. du Boulay, R. Mizoguchi (Eds.) *8th World Conference on Artificial Intelligence in Education (AIED'97)*, pp. 55-62.
- Suthers, D.D. Toth, E. E. and Weiner, A. (1997). An integrated approach to implementing collaborative inquiry in the classroom. In *Proc. of the 2nd International Conference on Computer Supported Collaborative Learning (CSCL'97)* (pp. 272-279). Toronto.