

REPORT DOCUMENTATION PAGEForm Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave Blank)		2. REPORT DATE 10 May 2004	3. REPORT TYPE AND DATES COVERED Final (15 Dec.2001 - 31 Dec. 2003)	
4. TITLE AND SUBTITLE LANGUAGE ANALYSIS AND GENERATION IN ALGEBRA TUTORIAL DIALOGUES FOR LANGUAGE-BASED INTELLIGENT TUTORING SYSTEMS			5. FUNDING NUMBERS 02PRO4143-00 N00014-02-1-0164	
6. AUTHORS Dr. Jung Hee Kim				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) North Carolina Agricultural and Technical State University 1601 E. Market St. Greensboro, NC 27411			8. PERFORMING ORGANIZATION REPORT NUMBER FRS # 4-41204	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of Naval Research Dr. Susan E. Chipman 800 North Quincy Street Arlington, VA 22217-5660			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) The North Carolina A&T State University algebra tutoring dialogue project collects and analyzes algebra tutoring dialogues with the aim of describing tutoring strategies and language with enough rigor that they may be evaluated and incorporated in machine tutoring. The tutoring is computer-mediated chat-style, with the tutor and student collaboratively solving a problem and editing equations. Transcripts of these dialogues are annotated for tutorial intentions, language, and behaviors. Comparison of tutoring by more and less expert tutors allows us to focus on the strategies that are most effective. The most prominent we observe is a metacognitive strategy where the tutor tries to elicit from the student a categorization of the problem under discussion and to name a solution method. We combine these strategies into a structured set of tutorial goals, with example sentences for each. In subsequent tutoring the computer then suggests these goals and sentences to the human tutor, who can pick use them in the dialogue. Measuring how much the tutors follow the computer-assistance, as well as student learning gains, allows us to evaluate the validity of our structural analysis of tutoring.				
14. SUBJECT TERMS Intelligent Tutoring Systems, Tutorial Dialogue Language Analysis Language Generation			15. NUMBER OF PAGES 17	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-1
298-102

20040513 048

NC A&T Algebra Tutorial Dialogue Project: ONR Grant Final Report

The goals of the NC A&T algebra tutorial dialogue project are to collect, analyze, and verify protocols from the human tutoring of algebra in a manner consistent with current work adapting human tutoring dialogues for intelligent tutoring systems.

1. Accomplishments Summary

With regards to collecting data, we have accomplished:

1. Recruited and tutored 62 students from January 2002 to November 2003.
2. Accumulated 51 keyboard tutoring transcripts, 5 video taped face-to-face sessions, and 55 sets of pre- and post-tests.
3. Used 1 expert tutor (24 transcripts), 2 experienced tutors (16 transcripts), and 1 novice tutor (11 transcripts).

With regards to data analysis, we have accomplished:

4. Completed a preliminary mark-up of all problems in 8 transcripts.
5. More fully analyzed 2 algebra problems across many transcripts, resulting in the identification of six tutorial methods and a number of associated topics.
6. Accumulated a preliminary collection of methods, topics, and language for one problem for the semi-automated Wozz tutor.
7. Analyzed learning gains, time per problem, number of problems worked on, number of turns per problem, etc. vs. student classroom achievement and tutor experience level (expert/experienced/novice).
8. Analyzed student initiatives in keyboard vs. spoken CIRCSIM-Tutor transcripts, and started applying this analysis to our algebra transcripts.

With regard to validating our markup we have accomplished:

9. Used the Wooz tutor to evaluate our analysis of the one problem by tutoring students.

With regards to software development, we have accomplished:

10. Developed the iTutor computer-mediated tutoring application, with ancillary software for researchers to navigate among, inspect, and print out human-readable transcripts.
11. Wrote the Wooz tutor, the computer-assist for human tutoring that validates our transcript analysis.

2. Approach

1. Collection of Transcripts

Students were recruited from the first-year algebra class at NC A&T State University. The students in this class are largely non-technical majors. The students were categorized by their teacher into one of five levels of academic achievement, based on their prior performance in the algebra class.

For our tutoring sessions, which were usually one hour long, the mathematics tutoring center at NC A&T provided a venue. Our expert tutor was Dr. Kathy Cousins-Cooper, a Ph.D. in mathematics education and co-PI on this grant, the two experienced tutors came from the staff of the tutoring center but have no formal training in mathematics education, and the novice tutor was an upper-level mathematics student.

Typical problems solved during these tutoring sessions include both symbolic manipulation problems and word problems, viz:

1. Please factor $3x^2 - 11x + 6$.

2. Bob drove “m” miles from Denver to Fargo. Normally this trip takes “n” hours, but on Tuesday there was good weather and he saved 2 hours. Write an equation for his driving speed “s”.

With every tutoring session we administered pre- and post-tests. The pre-tests were used for determining the highest priority topics for tutoring. The tutor picked those topics where the student gave the wrong answer, but did not leave the test sheet completely blank. We felt that these were the areas where the student was most likely ready to benefit from instruction. The post-test problems presented to a student were restricted to the problem areas that were tutored, so the learning gains recorded represent gains for the tutored topics only.

The keyboard-to-keyboard transcripts were collected using the iTutor computer-mediated tutoring program, constructed by this project. This software was later upgraded to become the Wooz tutor. Our transcripts were collected with the tutor and student in separate rooms so there could be no interaction during the session beyond what was captured by the computer.

We regard it as a high priority to put our transcripts in a usable, readable form, and make them available for dissemination to other researchers. To this end we developed software to make the transcripts readable, for example converting the MathML machine-readable representations of equations into normal printed representations, and interpolating the equations into the printed transcript of the dialogue stream at the points where the equations were updated. We also have software for navigating among transcript turns on-line.

In addition to the keyboard sessions, we have video tapes of five tutoring sessions with the expert and experienced tutors. These sessions have not been transcribed.

2. Finding Human Tutoring Tactics and Language

We have been performing intention-based markup of human tutorial dialogue, specifically with an aim to discover schemata that describe the tutor’s behavior. We are also interested in the

words and phrases the tutor uses, the observable trace of social interactions (e.g., the language the tutor uses to acknowledge various student responses), the topics the tutor chooses to teach and ignore, and other quantifiable linguistic phenomena such as use of discourse markers and styles of acknowledgments. Comparing transcripts prepared by expert and inexperienced tutors enables us to focus on the kinds of behaviors that make expert tutors better.

Our transcript analysis is in the same style as used by the CIRCSIM-Tutor project.

3. Categorizing Student Behaviors

In addition to categorizing the student answers to questions that drive the normal course of tutoring, we have been studying student conversational initiatives and the responses to open questions. Handling student initiatives and responses to open questions are an important goal at the edge of current research in dialogue-based tutoring.

4. Assessment of the Results

We have built a way to evaluate some of the tutoring strategies, a dummy machine tutor (the Wooz tutor) that constrains a human tutor to converse using the strategies and language we have coded from the transcripts. We have used two measures for evaluating the Wooz tutor. The first is the pre- and post-test results, which we can compare with the results from our unaided human tutors. The second comes from the fact that the human tutor vets and edits all language that the Wooz tutor emits, even if it was mechanically constructed. The extent to which the human tutor edits this language, or abandons the mechanically generated responses entirely, tells us how successful the Wooz tutor is being. Later, we can use survey instruments to find how the students and tutors react to Wooz tutoring sessions.

Since the Wooz tutor is an enhanced version of the iTutor software, tutoring sessions can use computer-aided tutoring for some problems and the regular tutoring for other problems all in the

same tutoring session. Thus we did not need to have markup and analysis of all problem types before we began Wooz tutoring.

3. Results

1. Tutoring Statistics

From pre- and post-tests we have measured learning gains during our tutoring sessions. These gains are calculated by $(\text{posttest} - \text{pretest}) / (100 - \text{pretest})$, where we selected from the pre- and post-tests only those topics that were tutored. Over all keyboard sessions, the learning gain averages 0.35. In the five face-to-face sessions, the learning gain averaged 0.19, less than in the keyboard-mediated sessions but there is not enough data to say this is significant. We conclude from these results that the tutoring we are studying is, in fact, having a strongly positive effect. This validates that the phenomena we extract from the transcripts should indeed be potentially useful. Furthermore, the expert tutor really is having more effect than the less experienced tutors, which means that the expert vs. novice analysis paradigm will focus on differences that are pedagogically significant.

Our students are divided into five ability groupings according to their classroom record prior to the tutoring. We find that the lower levels exhibit somewhat lower learning gains. The lower achieving groups need more time and turns to solve a problem compared to higher achieving groups, and solve fewer problems during one hour of tutoring. These results are summarized in Table1, Table2, and Table 3.

Learning gains were also correlated with tutor experience level. The expert tutor showed an average gain of 0.41 over 24 sessions, the experienced tutors' learning gain is 0.35 over their 16 sessions, and the novice tutor's learning gain is 0.18 over their 11 sessions.

Overall, students solved on the average of 5.1 problems per keyboard session, and 11 in the face-to-face sessions.

Table 1 Learning gains vs. student level

level	avg. pre test	avg. post test	difference	learning gains
1 (18 stu)	14.98	41.25	26.27	0.31
2 (13 stu)	25.28	45.44	20.16	0.27
3 (9 stu)	29.59	62.40	32.81	0.47
4 (6 stu)	43.07	63.10	20.03	0.35
5 (9stu)	52.78	76.54	23.77	0.50
Average	29.05	53.87	24.82	0.35

Table 2 Average of needed time & # of turns per problem vs. student level

level	avg. needed time per problem	avg. # of turns per problem
1 (16 stu)	18:23.8	19.93
2 (12 stu)	15:27.6	18.53
3 (8 stu)	09:38.6	14.43
4 (7 stu)	13:03.1	11.93
5 (8stu)	09:11.6	10.44
Average	14:09.3	16.15

Table 3 Avg. # of tutored problems per session vs. student level

level	Avg. # of tutored problems per session
1 (16 stu)	4.38
2 (12 stu)	4.25
3 (8 stu)	6.75
4 (7 stu)	4.14
5 (8stu)	6.88
Average	5.08

2. Transcript Analysis

Tutorial Goals and Language. From the transcripts we have learned quite a bit about the structure of our algebra tutoring. The most common structure is to 1) make sure the student is aware of what operation is being asked, 2) make sure the student is aware of the approach to take, and 3) make sure the problem is solved.

Figure 1 shows an extract from a relatively short dialogue where the student solved one multiplication problem. (In printed transcripts, the evolving equation in the equation window is interpolated into the dialogue every time the equation changes.) It well illustrates the strategy of ensuring that the student can state the type of problem (multiplying polynomials in this case) and a technique to solve it (a mnemonic device in this case). This tactic alone is often enough to get the student to solve the problem, as illustrated, even though the student failed to solve similar problems on the pre-test. Getting the student to explicitly state the problem and method is consistent with the view that learning mathematics often invokes metacognitive processes for just these sorts of operations.

Problem: Solve $(3x-2)(x+5)$

T1-tu-101-1: What operation do you need to apply here?

T1-st-102-1: Factoring.

T1-tu-103-1: No, Not factoring.

T1-tu-103-2: The expression is already factored.

T1-tu-103-3: What is the opposite of factoring?

T1-st-104-1: Distributing.

T1-tu-105-1: You can say distributing.

T1-tu-105-2: But we are multiplying here.

T1-tu-105-3: Now, what process do we use when we are multiplying two binomials.

T1-st-106-1: Foil.

T1-tu-107-1: Excellent, You use FOIL.

T1-tu-107-2: What does FOIL stand for?

T1-st-108-1: First Outer Inner Last.

T1-tu-109-1: Good, Now go ahead and use FOIL to multiply the two binomials together.

T1-st-110-1: $3x^2 + 15x - 2x - 10$

T1-st-111-1: $3x^2 + 13x - 10$

T1-tu-113-1: Excellent.

Figure 1: Excerpt of tutoring dialogue for multiplication problem

Each goal has subgoals, usually involving eliciting some responses from the student, that we call tactics. Some of the tutoring tactics we have identified and annotated are: “tutoring via

FOIL” (a mnemonic for multiplying binomials), “**correcting via factors**” (having the student factor individual coefficients of a polynomial), “**moving forward**” (letting the student proceed to see what happens), “**making binomials**” (mechanically setting up the form of the result, for the student to fill in), and “**trial and error**” (having the student guess a result and verify whether it is correct). Figure 2 shows the markup of a more substantial tutoring episode, illustrating these tactics.

Problem: factor $2x^2 - 7x + 6$

```

<T-identifies-operation>
  <T-tutors-operation>
    <T-elicits DM=now>
      T: Now what are we doing here?
      <S-ans catg=incorrect>
      S: Solving the quadratic equation. </S-ans>
    </T-elicits>
  </T-tutors-operation>
  <T-tutors-expression>
    <T-elicits>
      T: Is this an equation?
      <S-ans catg=correct>
      S: no, because there isn't an equal sign. </S-ans>
      <T-ack type=positive>
      T: That's right. </T-ack>
    </T-elicits>
    <T-informs>
      T: This is an expression.
    </T-informs>
  </T-tutors-expression>
  <T-tutors-operation attempt=2>
    <T-elicits DM=so>
      T: So what are we doing with this expression?
      <S-ans catg=correct>
      S: factoring. </S-ans>
      <T-ack type=positive>
      T: Good </T-ack>
    </T-elicits>
  </T-tutors-operation>
</T-identifies-operation>

```

Figure 2: Excerpt of annotated transcript

The goal structure of this dialogue can be summarized as follows:

Goal Hierarchy	Comment
IDENTIFY OPERATION	<i>Operation is factoring</i>
TUTOR OPERATION	<i>Elicit the operation from the student, 1st attempt</i>
TUTOR EXPRESSION	<i>Student did not recognize it is an expression, this is a fixup</i>
TUTOR OPERATION	<i>2nd attempt</i>
IDENTIFY APPROACH	<i>Solution approach is trial-and-error factoring</i>
TUTOR TRIAL-AND-ERROR	<i>Elicit from or inform the student to factor by trial-and-error</i>
SOLVE PROBLEM	
MAKE BINOMIALS	<i>Ask the student to produce both binomial terms all at once.</i>
OBTAIN FACTORS	<i>Student failed, obtain terms one at a time, 1st attempt</i>
OBTAIN FIRST FACTOR	
OBTAIN SECOND FACTOR	
CONFIRM FACTORING	
CHECK-VIA-FOIL	<i>Confirm by multiplying back The student made an error.</i>
OBTAIN FACTORS	<i>2nd attempt</i>
OBTAIN SECOND FACTOR	<i>Correct the one term the student got wrong</i>
(CONFIRM FACTORING)	<i>Student did this without the tutor prompting</i>

Following the identification of schemata, we collected examples of language used for each goal. The set of goals and associated sentences are then collected together, one set for each problem. Some of the sentences are simple templates where the variable slots can be filled in with the student's name or problem-specific information. An extract from the set of goals for the factoring problem, with a few of the sentences identified for each goal, is shown in Figure 3. This data is what the human tutor picks from when tutoring using the Wooz tutor.

<p>IDENTIFY THE PROBLEM <i>What are we doing with this expression?</i> <i>We are factoring.</i></p> <p>INTRODUCE THE TRAIL-ERROR APPROACH. <i>What method can we use to factor this?</i> <i>Let's use the trial and error method.</i></p> <p>SOLVE PROBLEM</p> <p>SOLVE VIA MAKING BINOMIALS</p> <p>MAKE BINOMIALS <i>So you set up your 2 sets of parentheses and fill them in with factors of the first term and the last.</i> <i>This trinomial will factor into 2 binomials</i></p> <p>CONFIRM FACTORING <i>Now how do we check our factors.</i> <i>Now use FOIL to check this</i></p> <p>MAKE FIRST TERM <i>What are the two factors of $2x$ squared?</i> <i>Ok, x squared factors into x and x, right?</i></p> <p>...</p> <p>SOLVE VIA CORRECTING FACTORS</p> <p>CORRECTS 1ST TERM</p> <p>CORRECTS 2ND TERM</p> <p>CORRECTS FACTOR SIGN</p> <p>...</p> <p>SOLVE VIA MOVE FORWARD <i>Go ahead and solve it</i></p>

Figure 3: Excerpt from goals and associated sentences for factoring problem

Student Initiatives. Our analysis of student initiatives started with a comparison of student behavior between keyboard and in-person tutoring in CIRCSIM-Tutor transcripts. Our data shows that while the initiatives are similar in some categories, the differences in the communication medium and conversational setting result in considerable differences both in kinds and numbers of student initiatives. In the face-to-face verbal transcripts, there were more student requests for information and more tutor interruptions of the student. Our preliminary analysis of the algebra keyboard-to-keyboard transcripts shows that they are similar to the CIRCSIM-Tutor keyboard-to-keyboard ones in these regards.

3. Evaluating Transcript Markup with the Wooz Tutor

The binomial factoring problem was the only problem sufficiently analyzed that we could evaluate our analysis with the Wooz tutor. The machine tutor provided assistance to the human tutor on this problem during six tutoring sessions, compared with fourteen sessions where the problem was tutored without machine assistance.

In Wooz tutor operation, the human tutor observes a tree-structured map of dialogue goals. Clicking on a goal reveals a menu of sentences that the tutor can pick from. These sentences are then subject to editing before the tutor submits them into the dialogue. From the student's viewpoint, there is no difference between the computer-assisted dialogue and the regular dialogue.

The average learning gain on the one Wooz-tutored problem type was 0.75 for the computer-assisted tutoring (six cases) and -0.16 (a loss) for the sessions where the same problem received unassisted tutoring (15 cases). We noticed that the six Wooz-tutor sessions were all at the lower 3 of our 5 academic achievement levels, which affects expected learning gain. Matching on student achievement level, the unassisted tutoring of this same problem (10 cases) exhibits a learning gain of 0.0.

Table 4. Wooz-tutoring vs. unassisted human tutoring

	Wooz-Assisted Dialogues	All Manual Dialogues	Achievement-Matched Dialogues
Learning Gain	0.75	-0.16	0.00
Avg. Turns	21.83	20.13	25.30
Avg. Time	18:52	14:05	18:04
N	6	15	10

It is not clear to us why the Wooz computer-assisted tutoring was so much better than unassisted tutoring. There is not enough data to get statistical significance, so we can not be sure the increased learning gains are a reproducible effect. We speculate, however, that in the process of examining the transcripts, particularly with the focus on this one problem, the tutors improved their performance on the problem type in question. This effect would also explain why the face-to-face transcripts exhibited less of a learning gain than the keyboard-to-keyboard transcripts, as they were all collected and examined at the start of the project.

The other way to evaluate the markup was to observe how frequently the human tutor edited or replaced the Wooz tutor-suggested language before submitting it to the student. In fact, the tutors almost always followed the suggested tutorial goal schemata. This suggests that we have the goal structure correct. We have not tried the computer-suggested goal structure and dialogue with novice tutors to see whether it affects their tutoring.

Of the tutor turns in the Wooz-assisted dialogue, 70% were extracted from the database of canned sentences with no change, 6% were edits of existing sentences, and 24% were new sentences. There seems little observable qualitative difference between the edits and the new sentences, it seems that once the tutor started editing a sentence she changed almost the whole thing. The new and changed sentences were still in service of the same tutorial methods. However the new and changed sentences almost always respond to specifics of student utterances that did not appear in the attested transcripts used in building the sentence database. Here is an example of a modified turn:

Stu: *I'm going to use the quadratic formula.*

Tut (original): *Is this an equation?*

Tut (edited): *We use the quadratic formula for quadratic equations. Is this an equation?*

This phenomenon, the human tutor responding to specific variations in the student responses, would seem to reduce the Wooz tutor's evaluative probity. When a tutor changes a sentence, we have no way to know whether the unchanged sentence would have worked just as well. Nevertheless, with experience we should build up knowledge of what rates of sentence modifications to reasonably expect.

4. Software Application Development

One result of this project is that we have developed software that will be of use to us in the future and possibly to other similar projects, the iTutor/Wooz collection of programs for computer-mediated keyboard to keyboard tutoring of algebra. These programs support a model of tutoring where the tutor and student collaboratively solve algebra problems. The equations being worked on, the dialogue, and the original problem statement, are visible in separate windows. Both the tutor and student can edit the equation and contribute the dialogue.

In more detail, the tutor and student use almost identical screens, illustrated in Figure 4. The "load question" button appears on the tutor's side only, enabling the tutor to pick from a collection of pre-canned problems which are loaded into the problem statement and equation areas. Each question customizes the equation-editing toolbar, deactivating mathematical operations that are not useful for this problem. The dialogue is interleaved in a chat window. During the dialogue, the two parties cannot type simultaneously, so the application must keep track of who is taking a turn. The tutor can seize the turn-taking state, interrupting the student, but not vice versa. After a turn, the software returns to a quiescent state, so either party can continue the conversation.

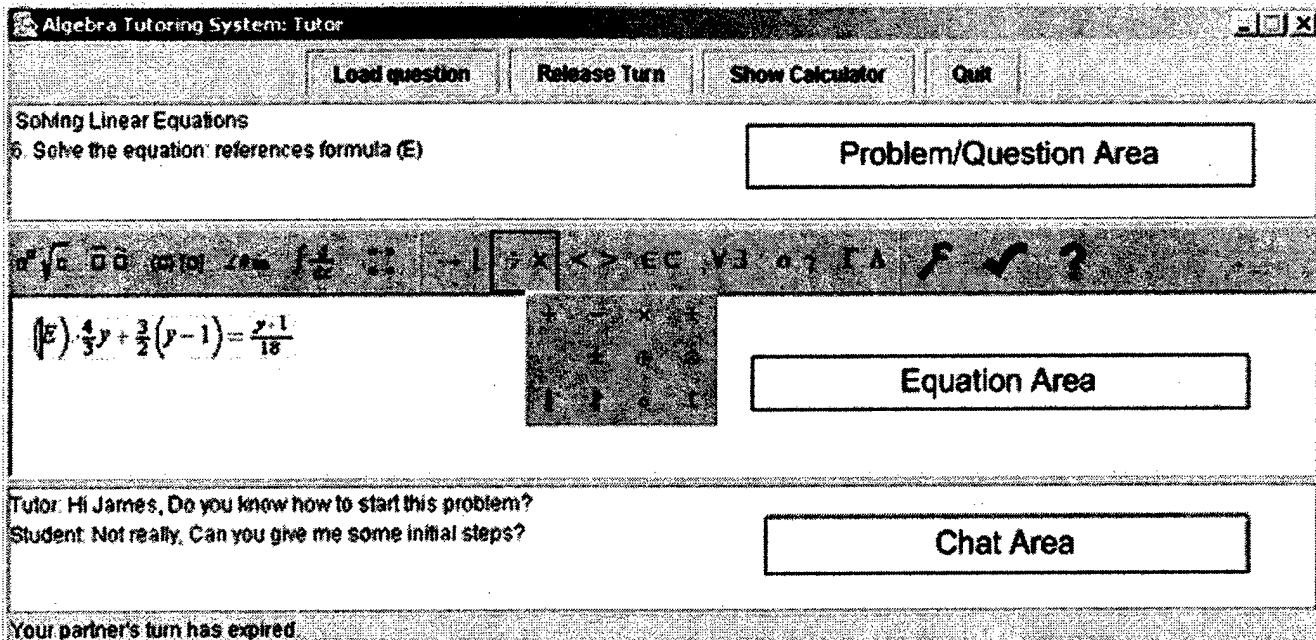


Figure 4: iTutor client interface

The Wooz tutor upgrade to the basic iTutor application works as follows. From the student's screen, it looks the same as iTutor for computer-mediated human tutoring. However on the tutor's screen there is a tree structure of tutoring goals. Expanding a node on the tree yields a selection of sentences the tutor can pick. Some of the sentences are automatically filled in with information from the current tutoring environment, for example parts of the problem equation or the name of the student. (Some of our tutors occasionally address the students by name. Since this is possibly a significant motivational technique, we do not want to exclude that behavior from our packaged tutoring protocols.) Other parts of the chosen sentence can be filled in as needed, for example discourse cue words and acknowledgments to the student. The whole sentence can be edited or replaced before the tutor sends it to the student. The pre-packaged sentence is almost always picked from attested sentences in the transcripts. The tutors' job is to

navigate through the tree hierarchy, picking alternatives or retrying, or even abandoning the structure, as they see fit.

The iTutor/Wooz software application is developed in JAVA™ using a client-server architecture. The same client program runs on two different computers in two different rooms, one for the tutor and one for the student. The server program runs on potentially yet a third computer, managing communication between the tutor and student screens and logging all interactions in disk files. All communication is via TCP/IP, and the user interface screens are written with standard Java interface classes, so the software is largely platform-independent.

Among the ancillary software is a transcript reader program, and software for turn-numbering and printing in readable form. The log reader allows the user to navigate and inspect conversations. Among its features is an ability to display the differences between successive versions of an equation, so the user can observe what in particular the student is changing.

For the Wooz tutor, we also have ancillary software for gathering sentences and goal structures into a database, which is accessed during tutorial operation.

4. Dissemination and Continuation

Transcripts of our tutorial dialogues will be submitted to repositories of such transcripts, for example the collection run by the Circle project at LRDC which already contains Neal Heffernan's algebra tutorial dialogues.

We have submitted a proposal to the Education Department's MSEIP program to use our project's facilities and expertise for improving the education of science and engineering students. The idea is to train science and engineering students to tutor elementary algebra, thereby improving their own skills in communication and training and exercising their metacognitive abilities in mathematics. To this end, they would use our tutoring setup: the iTutor/Wooz

interface, the pre- and post-tests, the mathematics tutoring center, and project personnel. The students would then be able to 1) review in detail (with experts) transcripts of their own tutoring, 2) measure their performance, and 3) inspect the structure of known-good tutoring interactions.

We have submitted several proposals, with no success to date, to continue and expand the work described here.

5. Statistics

Number of Degrees Granted : Two Master of Science degrees (Niraj Patel, Jie Zhao)

Number of Undergraduate Students employed: Four

PI/CoPI Minority Women: One as a Co-PI (Dr. Kathy Cousins-Cooper)

PI/CoPI Non-Minority Women: One as a PI (Dr. Jung Hee Kim)

6. Journal Articles

Kim, J.H., Freedman, R, Glass, M., and Evens, M.W. (submitted 2003) Annotation of Tutorial Goals for Natural Language Generation. submitted to Discourse Processes.

7. Technical Reports

Patel, Niraj, Michael Glass, Jung Hee Kim. (2003) Data Collection Applications for the NC A&T State University Algebra Tutoring Dialogue (Wooz Tutor) Project. In Proceedings of the Fourteenth Midwest Artificial Intelligence and Cognitive Science (MAICS-2003) Conference, Cincinnati, OH, pp.120-125.

Zhao, Jie, Jung Hee Kim, Martha W. Evens.(2003) Comparison of Student Initiatives in Keyboard-to-Keyboard and Face-to-face Tutoring Sessions. In Proceedings of the Fourteenth Midwest Artificial Intelligence and Cognitive Science (MAICS-2003) Conference, Cincinnati, OH, pp178-183.

Purrington, Kevin, Jung Hee Kim. (2003)iTutor: Software for Computer-Mediated Algebra Tutoring. In Proceedings of the Seventh NC-LSAMP Conference, Pembroke, NC.

Kim, Jung Hee, Michael Glass, Evaluating Dialogue Schemata with the Wizard of Oz Computer
–Assisted Algebra Tutor, submitted to Intelligent Tutoring System 2004 conference

Cousins-Cooper, Kathy, Jung Hee Kim, Michael Glass, Human Tutoring Tactics for Computer
Tutoring Systems in Algebra, submitted to Mid-Atlantic Consortium for Research in
Mathematical Sciences (MCRMS) 2004 Conference