AutoTutor: An Intelligent Tutoring System With Mixed-Initiative Dialogue

Arthur C. Graesser, *Fellow, IEEE*, Patrick Chipman, Brian C. Haynes, and Andrew Olney

***Abstract*—AutoTutor simulates a human tutor by holding a conversation with the learner in natural language. The dialogue is augmented by an animated conversational agent and three-dimensional (3-D) interactive simulations in order to enhance the learner's engagement and the depth of the learning. Grounded in constructivist learning theories and tutoring research, AutoTutor achieves learning gains of approximately 0.8 sigma (nearly one letter grade), depending on the learning measure and comparison condition. The computational architecture of the system uses the .NET framework and has simplified deployment for classroom trials.**

***Index Terms*—Conversational agents, intelligent tutoring systems, natural language dialogue, STEM learning, tutoring.**

AUTOTUTOR is a complex system that simulates a human or ideal tutor by holding a conversation with the learner in natural language [1]–[3]. AutoTutor presents a series of challenging questions (or problems) that require approximately a paragraph of information to answer correctly. An example question in conceptual physics is, "When a car without head rests on the seats is struck from behind, the passengers often suffer neck injuries. Why do passengers get neck injuries in this situation?" Although a perfect answer to this question is approximately three to seven sentences in length, the initial answers by actual human learners are typically only one word to two sentences in length. AutoTutor assists the learner in the construction of an improved answer that draws out more of the learner's knowledge and that adaptively corrects problems with the answer. The dialogue between AutoTutor and the learner typically lasts 50 to 200 conversational turns for one challenging question. Table I presents an annotated analysis of the example dialogue by specifying the categorized dialogue moves of AutoTutor, the classified speech acts of the student, and assorted comments to help the reader interpret the tutorial dialogue.[1]

Manuscript received July 29, 2004; revised July 9, 2005. The research on AutoTutor was supported by the National Science Foundation (SBR 9720314, REC 0106965, REC 0127275, ITR 0326428) and the DoD Multidisciplinary University Research Initiative (MURI) administered by ONR under Grant N00014-00-1-0600. The Tutoring Research Group (TRG) is an interdisciplinary research team comprising approximately 35 researchers from psychology, computer science, physics, and education (see http://www.autotutor.org). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF, DoD, or ONR.

The authors are with the University of Memphis, Memphis, TN 38152-3230 USA (e-mail: a-graesser@memphis.edu).

Digital Object Identifier 10.1109/TE.2005.856149

[1]The content expressed by either AutoTutor or the student in Table I are signified in italics. Discourse categories of AutoTutor's dialogue moves have been added in capitals, whereas other information is added in normal font.

This paper has four parts. First, there is a brief overview of AutoTutor's pedagogical strategies that embody a constructivist approach to instruction. Second, student assessment data are presented from both controlled laboratory experiments and classroom use of AutoTutor. Third, the architecture of the AutoTutor intelligent tutoring system is presented. Finally, a description is given of some new advancements in the system, including a version of AutoTutor on conceptual Newtonian physics that has interactive simulations.

I. Pedagogical Foundations of AutoTutor

The design of AutoTutor was motivated by three lines of research in education and cognitive science. First, there are explanation-based constructivist theories of learning [4], [5]. Such theories assume that learning is more effective and deeper when the learner must actively generate explanations, justifications, and functional procedures than when merely given information to read. The explanations are pedagogically deep in the sense that the learner must be able to express the causal and functional underpinnings of a mechanism rather than mechanically applying procedures and formulae. Second, there are intelligent tutoring systems that adaptively respond to the learner's actions [3]–[6]. These tutors give immediate feedback to the learner's actions and guide the learner on what to do next in a fashion that is sensitive to what the system infers the learner knows. Third, empirical research has documented the collaborative constructive activities that routinely occur during human tutoring [7], [8]. The patterns of discourse uncovered in naturalistic tutoring were imported into the dialogue management facilities of AutoTutor.

One of AutoTutor's prominent dialogue patterns is called *expectation and misconception tailored dialogue* (EMT dialogue), which is known to be common in human tutoring [3], [8], [9]. Both AutoTutor and human tutors typically have a list of anticipated good answers (called *expectations*) and a list of *misconceptions* associated with each main question or problem. One goal of the tutor is to coach the student in covering the list of expectations. Another goal is to correct misconceptions that are manifested in the student's talk and actions. Yet another goal is to give feedback and adaptively respond to the student. The expectations and misconceptions associated with a main question are stored in AutoTutor's *curriculum script*. AutoTutor provides *feedback* to the learner (positive, neutral, and negative feedback), *pumps* the learner for more information ("What else?"), *prompts* the learner to fill in missing words, gives *hints*, fills in missing information with *assertions*, identifies and *corrects* bad answers, *answers* learners' questions, and *summarizes* answers.

TABLE I
EXAMPLE AUTOTUTOR–STUDENT DIALOGUE WITH CLASSIFIED DIALOGUE MOVES OF AUTOTUTOR

AutoTutor-1: *When a car without headrests on the seats is struck from behind, the passengers often suffer neck injuries. Explain why.* (MAIN QUESTION, This is a main question selected from AutoTutor's curriculum script, which will take approximately 50-200 conversation turns to answer collaboratively.)

Student-2: *I don't know.* (This expression by the student is classified as a Metacognitive speech act because it conveys what the student knows about his/her own knowledge.)

AutoTutor-3: *Could you at least try to give me an answer? That way, we'll have something as a starting place to work from.* (PUMP. This dialogue move encourages the student to say more, but does not direct the student to express particular information.)

Let's try this together. (DISCOURSE MARKER. This is a discourse marker that connects different dialogue moves within a turn. Another discourse marker is "Okay.")

When the neck provides a large force to accelerate the head, what could happen to the neck? (HINT. This dialogue move leads the student to articulate a particular sentence, clauses, or proposition.)

Student-4: *What is a force?* (The student can ask questions during an AutoTutor session.)

AutoTutor-5: *I'll take a crack at answering your question.* (DISCOURSE MARKER

Force is any influence that tends to accelerate an object; a push or pull; measured in newtons. Force is a vector quantity. (ANSWER. AutoTutor answers the student's direct question.)

Was that what you were looking for? (QUESTION SOLICITATION. AutoTutor allows the student the opportunity to rephrase his question.)

[At this point, the dialogue continues until AutoTutor believes it has detected a misconception, as shown in the following excerpt.]

AutoTutor-20: *See if you know the answer to this.* (DISCOURSE MARKER. AutoTutor prepares the student for a direct question.)

When the head and body are moving at the same acceleration, they are moving at the same ________? (PROMPT. This dialogue move encourages the student to fill in a particular word or phrase of information, in this case the word "rate." Prompts are needed for students who are low verbal and to elicit a particular noun, verb, or adjective.)

Student-21: *distance*

AutoTutor-22: *Moving at the same rate.* (PROMPT COMPLETION. AutoTutor splices in the correct information if the student provides an incorrect answer to its prompt.)

[Several more turns take place until all expectations are covered.]

AutoTutor-136: *Okay, now, let's recap.* (DISCOURSE MARKER

In the collision notice the person's body and head move along together. In order for this to occur the body and head must have the same acceleration. This concept applies to all situations where two bodies are moving together through space. (SUMMARY. AutoTutor recaps what the student has learned.)

Now, let's go onto another physics problem. (DISCOURSE MARKER. AutoTutor selects another problem to work on.)

As the learner expresses information over many turns, the information in the three to seven sentences of an expected answer is eventually covered, and the question is scored as answered. During the process of supplying the ideal answer, the learner periodically articulates misconceptions and false assertions. If these misconceptions have been anticipated in advance and incorporated into the program, AutoTutor provides the learner with information to correct the misconceptions. Therefore, as the learner expresses information over the turns, this information is compared with expectations and misconceptions, and AutoTutor formulates its dialogue moves in a fashion that is sensitive to the learner input. In short, the dialogue management component adaptively selects hints, prompts, assertions, and targeted three-dimensional (3-D) simulations in an attempt to achieve pattern completion.

A. Evaluating Student Contributions

AutoTutor uses Latent Semantic Analysis (LSA) as its primary, conceptual, pattern-matching algorithm when evaluating whether student input matches the expectations and misconceptions. LSA is a high-dimensional statistical technique that measures the conceptual similarity of any two pieces of text [9], [10]. The size of the text may vary from being a single word to a sentence, paragraph, or lengthier document. LSA-based similarity metrics are capable of evaluating the quality of learner contributions almost as well as graduate students in the subject matter, i.e., physics or computer literacy [1], [9]. With LSA, students can use bad grammar and vague, approximate terminology as long as the words that they use are associated with the correct words in the LSA space. This use of LSA makes AutoTutor robust and domain portable.

B. Dialogue Mechanisms and Strategies

AutoTutor's style of pedagogy is captured computationally in several mechanisms. First, the order of introducing main questions and covering expectations is not fixed or linear, but rather follows different trajectories depending on the talk and actions of the student. By allowing students to follow their own natural trajectory through the curriculum content, AutoTutor promotes the active generation of explanations according to the constructivist pedagogy. Second, a mechanism for AutoTutor selects hints, prompts, assertions, and other dialogue moves that attempt to maximize the student articulating a particular expectation or a lengthy answer. The mechanism is somewhat analogous to the minimax algorithm in chess-playing programs, except in this case AutoTutor tries to maximize the student's scoring points in contributing information, instead of minimizing it.

Third, AutoTutor uses and tests alternative pedagogical principles when deciding what expectation to select next during the course of coaching the student to construct explanations. Several of these principles [1], [11], [12] exist, but only two are addressed here. The *frontier learning* or *zone of proximal development* principle selects the next expectation that builds on what the student knows. An example will explain how this principle is implemented in AutoTutor. For example, there may be seven expectations that have LSA coverage values that vary from zero to one. These LSA values increase as the AutoTutor–learner conversation evolves, and more expectations get covered. At some point, two of the seven may have been covered because the LSA coverage score meets or exceeds some threshold T; the other five expectations have subthreshold coverage values. The frontier learning principle selects the expectation with the highest subthreshold LSA value. This decision mechanism is different from the *coherence* principle, which attempts to maximize the coherence of the conversation. The coherence principle selects the next expectation that is most similar to the previous expectation that was just covered; the similarity is once again determined by the LSA.

The dialogue mechanisms of AutoTutor are both computationally manageable and similar to what human tutors do. Human tutors cannot deeply comprehend all of the contributions of students, most of which are imprecise, vague, fragmentary, incomplete, and ungrammatical [8]. What human tutors can do is compare student input with anticipated good answers and misconceptions. LSA provides a suitable algorithm for these comparison operations.

II. Empirical Evaluations of AutoTutor

Four performance criteria have been considered in previous evaluations of AutoTutor. The first is whether particular computational modules of AutoTutor produce output that is valid and meets the intended technical specifications. Prior analyses have shown that AutoTutor's LSA component performs conceptual pattern-matching operations almost as well as human judges [9]. When AutoTutor grades the percentage of expectations that are covered in an answer to a question, the correlation between AutoTutor and a graduate student expert has been approximately $r = .50$, whereas two graduate students correlate approximately $r = .60$. AutoTutor's speech act and question classifier have also had a high degree of accuracy [13], with between .86 and .97 of its decisions agreeing with experts of language and discourse. The second type of evaluation assesses the quality of the dialogue moves produced by AutoTutor. The central question is whether AutoTutor's dialogue moves support a conversation that is coherent and relevant. The third criterion is whether AutoTutor produces learning gains. The fourth is whether learners enjoy interacting with AutoTutor. The scope of this paper does not permit a review of all of the performance evaluations of AutoTutor, but a few words are devoted to the second and third types of evaluation.

A. Evaluations of Dialogue Quality

Expert judges have evaluated AutoTutor with respect to conversational smoothness and the pedagogical quality of its dialogue moves [1], [2], [14]. Perhaps the most informative test has been a *bystander Turing test* on the naturalness of AutoTutor's dialogue moves [14]. In these studies, a random selection of tutor moves in the tutorial dialogues between students and AutoTutor takes place. Human tutors are then asked to fill in what they would say at these random points. Therefore, at each of these random tutor turns, content is generated by either AutoTutor or human tutors. A group of students (the bystanders) are subsequently asked to decide whether these dialogue moves are generated by a computer versus a human; half, in fact, were generated by computer and half by humans. The results surprisingly revealed that the bystander students were unable to discriminate whether particular dialogue moves had been generated by a computer versus a human. The hit rate did not significantly differ from chance, .51, when computing the likelihood that bystanders decided that a computer produced the turn when, in fact, the computer had; the false alarm rate was .53 (i.e., bystanders deciding a computer produced a turn when, in fact, a human did). The participants could not distinguish AutoTutor from a skilled human tutor when rating moves on a more sensitive six-point scale that varied from "likely a computer" to "likely a human"; in other words, the effects of changing the speaker were not significantly detectable in any statistical tests, $\mathrm{F}(1, 140) = 0.07$, $\mathrm{p} > .50$. These results of the bystander Turing test support the claim that AutoTutor is a good simulation of human tutors. AutoTutor manages to have productive and reasonably smooth conversations even though it approximately understands, but does not completely understand, what the student expresses. Perhaps tutorial dialogue is not highly constrained; thus, the tutor has a high degree of latitude on what can be said without disrupting the conversation. In this sense, tutoring conversations are more flexible than other conversation settings.

B. Evaluations of Pedagogical Quality

AutoTutor has been evaluated on learning gains in over a dozen experiments on the topics of computer literacy [1] and conceptual physics [15]. These studies have been conducted on college students in either a laboratory or a classroom setting. In some experiments, the students have intermediate-level knowledge that is achieved shortly after they are being trained on Newtonian physics. These students received extra course credit either 1) at the beginning of a second undergraduate physics course or an engineering course that covered Newton's laws or 2) in an initial college course at a point in the curriculum immediately after Newtonian physics is covered. In other experiments, the students were enrolled in an introductory computer literacy course where they received extra credit for participating at a point in the curriculum at or near the particular AutoTutor content that was covered (hardware, operating systems, and Internet). More recent studies have allowed physics and engineering students to complete the AutoTutor studies at home on the Internet.

The results of these studies have been quite positive, but before discussing such data, evaluations of learning gains must be placed in some perspective. The literature in the area has established that one-to-one human tutoring is a powerful method of promoting learning [16], [18], even though most human tutors have moderate subject matter knowledge and no training in pedagogy or tutoring. These unaccomplished human tutors enhanced learning with an effect size of .4 standard deviation units (called sigmas), which translates to an improvement of approximately half a letter grade [16]. Intelligent tutoring systems with sophisticated pedagogical tactics, but no natural language dialogue, produce effect sizes of approximately 1 sigma [18]. Effect sizes are focused upon here because sigmas have recently been adopted as the appropriate statistical yardstick in assessing the impact of treatments in education research. Traditional tests of statistical significance (e.g., t or F) do not measure the magnitude or robustness of treatment effects, but only the likelihood that any significant effect at all (binary decision) exists, independent of the magnitude of the effect.

The methodology of testing the impact of AutoTutor on learning gains is straightforward. Learners complete a pretest on the subject matter, and then interact with AutoTutor (for one to eight hours, depending on the experiment), and then complete a post-test that is an alternate counterbalanced version of the pretest. The AutoTutor treatment is compared with one or more comparison conditions that remove one, many, or all pedagogical components of AutoTutor. In ten experiments on over 1000 college students, AutoTutor has almost always produced statistically significant gains of .2 to 1.5 sigma (with a mean of .8), depending on the learning performance measure, the comparison condition (either pretest scores or a control condition in which the learner reads the textbook for an equivalent amount of time as the tutoring session), the subject matter, and the version of AutoTutor [1]. For example, a recent study involved testing AutoTutor in the physics domain on 35 college students enrolled at three universities. The participants were given AutoTutor training, allowed to read a textbook on physics, or given no training at all. A learning gain of 1.02 sigma over the textbook control was found, which was significant when analyzed using an ANCOVA with the pretest scores as a covariate, $\mathrm{F}(2, 31) = 11.37, \mathrm{p} < .01$ [15].

Approximately a dozen measures of learning have been collected in these assessments on the topics of computer literacy and physics. These include the following:

1) multiple-choice questions on shallow knowledge that tap definitions, facts, and properties of concepts;
2) multiple-choice questions on deep knowledge that taps causal reasoning, justifications of claims, and functional underpinnings of procedures;
3) essay quality when students attempt to answer challenging problems;
4) a cloze task that has subjects fill in missing words of texts that articulate explanatory reasoning on the subject matter;
5) performance on problems that require problem solving.

These evaluations place previous versions of AutoTutor somewhere between an unaccomplished human tutor and an intelligent tutoring system. Surprisingly, one recent evaluation of physics tutoring reported that the learning gains produced by accomplished human tutors in computer-mediated communication were equivalent to the gains produced by AutoTutor [15]. The largest learning gains have been on deep-reasoning measures rather than measures of shallow knowledge. Replications of these analyses on the most recent version of AutoTutor with interactive simulation (AutoTutor 3-D, as discussed later) are in progress and look promising.

To collect data in more real-world educational settings, AutoTutor has been made freely available on the Internet for any interested course instructor or student. Data have been collected at three southern universities other than the University of Memphis, Memphis, TN. For example, in fall 2004, 32 students from two classes completed AutoTutor and showed a statistically significant increase in learning between pretest to post-test ($\mathrm{sigma} = .6$, $\mathrm{p} < .05$). The distributed nature of the new AutoTutor architecture makes collecting data from large numbers of simultaneous users ($N = 500$) much more efficient than in the previous versions, resulting in a fiftyfold increase.

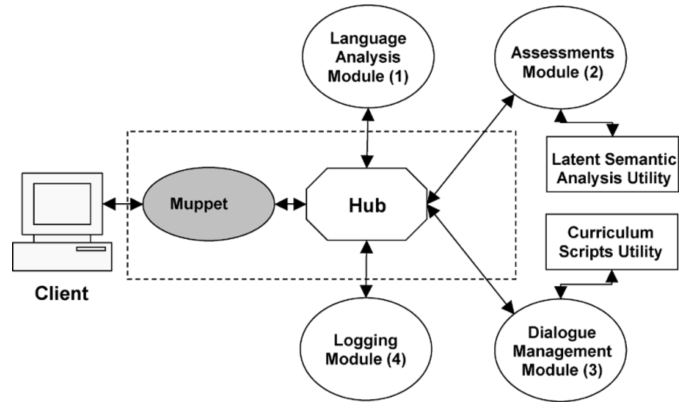

Fig. 1. Overview of the AutoTutor architecture.

III. Architecture of AutoTutor

This section briefly sketches the overall computational architecture of AutoTutor. Several versions of AutoTutor's architecture have been described in previous literature [2], [3], [17]; however, this paper focuses on the most recent version with interactive 3-D simulation. This version is called AutoTutor-3D, even though there is the option of removing the 3-D simulation module.

AutoTutor 3-D is a distributed client–server application on the Internet, capable of "scaling out" to multiple computers as necessary to handle greater load. The components of the system are divided into *modules, utilities,* and the *hub*, as illustrated in Fig. 1. The AutoTutor-3D server is a distributed *hub-and-spokes* application that may reside on multiple servers or on a single server. In the standard hub-and-spokes configuration, only the hub knows about the existence of the modules. Therefore, any component that provides the data specified by a *state object* may be interchangeably used. The hub receives a state object from the client and then passes the state object to various *modules* in a scripted order [17]. The basic architecture concept is similar to that of the DARPA Communicator [19]. Modules are defined by their ability to input and output state objects, which greatly simplifies communication and interoperability between modules. Modules call various utilities, such as LSA facilities or databases that have their own distinctive interface [17]. The AutoTutor-3D architecture is much like a production line in that modules each do a small bit of the work, and subsequent modules are dependent on preceding ones. Just as in a production line, the modules are only interested in the work in front of them and forget about previous work.

A concrete description of the processing stream is provided when AutoTutor-3D attempts to handle one conversational turn of the learner. The client computer sends a state object to the hub after the student has typed in the content of the turn. The state object is first passed to the language analyzer, which segments the utterance into main clauses, parses the clauses, and assigns a speech act to each main clause [13]. The language analyzer uses the Conexor EngLite parser [20] as a utility. The modified state object is sent to the hub, which then sends it to the assessor. The assessor updates the student model (i.e., what the student knows about the expectations and misconceptions associated with the problem) and produces a set of predictions about the likely effects of alternative dialogue moves on the student model. The assessor uses the LSA and the curriculum script database (i.e., the major content repository of main questions and dialogue moves) to update the student model and make predictions. Next, the assessor passes the updated state object to the hub, which forwards it to the dialogue manager. The dialogue manager consults the output of the previous modules and the dialogue information state of the previous turn. The dialogue manager subsequently updates the dialogue information state of the state object and provides it with dialogue for the tutor's turn. The updated state object is passed to the hub, which forwards it back to the client. The cycle is then completed for interpreting the student's turn and producing an adaptive response.

A few technical points are provided for those who are interested in implementing similar systems. AutoTutor-3D is written in C# and Visual Basic .NET, both languages that run atop the managed .NET framework and common language runtime (CLR). The CLR offers language-level compatibility [21], meaning that each component of AutoTutor can easily communicate with any other component, as long as the component is written in a CLR-compatible language. Over 30 programming languages have CLR-compatible compilers [22]; therefore, programmers can use the language of their choice. The .NET framework offers a variety of libraries, including the Remoting system that allows objects to be accessed quickly and transparently over the network and that forms the basis of the scalable infrastructure. Both the framework and the CLR virtual machine run on Windows, Linux, Mac OS X, and other operating systems, using open-source run times such as Mono and Portable .NET. The use of Remoting's binary communication channel allows the individual state table objects to be sent with minimal latency, using under 12 kB of bandwidth. The architecture's abstraction of the details of this interface provides developers with inherent thread safety and scalability. Current analyses in progress suggest that, by using these technologies, any particular instance of an AutoTutor server can handle over 500 simultaneous users with a memory footprint of about 180 MB [17].

IV. Recent Advances in AutoTutor

AutoTutor's extensible architecture is sufficiently powerful and flexible that additional advanced learning technologies can be quickly accommodated. Two of these advances are interactive 3-D simulations and responses to the learner's emotional states. The AutoTutor version with interactive 3-D simulation has been applied to conceptual physics and is currently being tested in physics and engineering courses.

A. AutoTutor With Interactive 3-D Simulation

The most recent version of AutoTutor has an embedded interactive 3-D simulation. A 3-D simulation provides an additional channel of communication to discuss conceptual physics with the learner. Each simulation is crafted to cover particular physics expectations or to help correct particular misconceptions. For each physics problem, an interactive simulation world in *3-D Studio Max* was developed, which included the people, objects, and spatial setting associated with the problem. The students can manipulate parameters of the situation (e.g., mass of objects, speed of objects, or distance between objects) and then ask the system to simulate what will happen. They can compare their expected simulated outcome with the actual outcome after the simulation is completed. Moreover, they describe what they see. Their actions and descriptions are evaluated with respect to covering the expected principles in an ideal answer. In order to manage the interactive simulation, AutoTutor gives hints and suggestions, once again scaffolding the learning process with dialogue. Thus, AutoTutor combines interactive simulation with mixed-initiative dialogue.

Fig. 2 shows the interface for the 3-D version of AutoTutor. The main question is presented at the top of the screen. Beneath the question are two windows that show the car and truck (middle window) and the driver in the car (right window). These components move whenever a simulation is run. Beneath the question on the left is the animated agent that guides the interaction with hints, suggestions, assertions, and other dialogue moves. These suggestions include having the student manipulate parameters, such as truck speed, mass of the car, and mass of the truck. The students also have a number of binary options: having the head rests in the car on, showing the skin on the driver, slowing down the simulation, and displaying vector arrows that depict forces. The student manipulates these parameters and options, as shown in the bottom left, before a simulation is run. The activity of manipulating these inputs and viewing the simulation is believed to provide a referentially grounded and embodied representation of the problem and a deeper understanding of physics. The students can run as many simulations as they wish until they feel they understand the relationship between the parameters and outcomes of simulations. However, interacting with and viewing the simulations is not all that is available. The participants are also prompted to describe what they see and answer the main question. Therefore, deep learning of physics is believed to emerge from the combination of interactivity, perceptual simulation, feedback on the simulation, and explaining what happens.

There are a small number of other learning systems with conversational agents that combine dialogue with interactive simulation. Examples of these systems are Steve [23] and Mission Rehearsal [24]. AutoTutor is the only system that is available on the Internet that has systematically been tested on students and that has flexible tutorial dialogue that scaffolds interactive simulation.

B. Adapting to Learner Emotions

A version of AutoTutor that perceives and responds to learner emotions in addition to the learner's knowledge states is currently in development. AutoTutor is augmented with sensing devices and signal processing algorithms that classify the affective states of learners. Emotions are classified on the

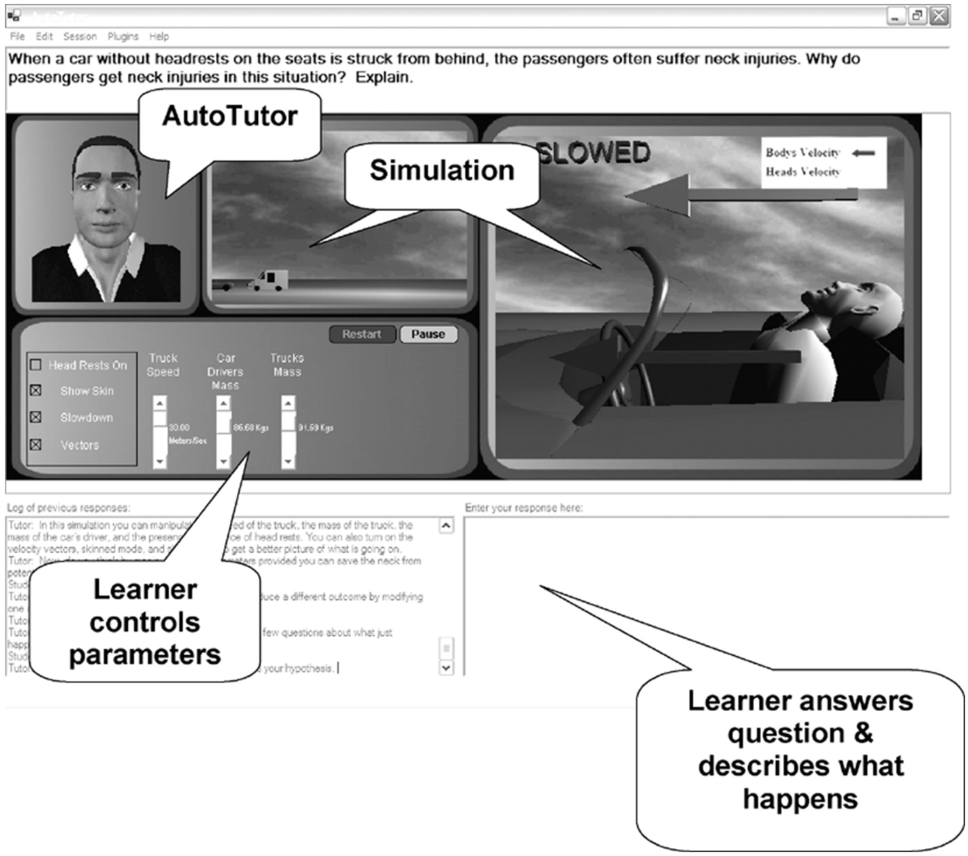

Fig. 2. Computer screen of AutoTutor on conceptual physics with interactive 3-D simulation.

basis of dialogue patterns during tutoring, the content covered, facial expressions, body posture, mouse haptic pressure, and keyboard pressure. This current project has two specific objectives. First, AutoTutor will analyze patterns of facial, body, and dialogue activity that arise while interacting with AutoTutor and will classify this input into basic affect states (such as confusion, frustration, boredom, interest, excitement, and insight). Second, empirical investigations will take place on whether learning gains and learner's impressions of AutoTutor are influenced by dialogue moves of AutoTutor that are sensitive to the learner's emotions. For example, if the student is extremely frustrated, then AutoTutor presumably should give a good hint or prompt that directs the student in a more positive learning trajectory. If the student is bored, AutoTutor should give more engaging, challenging, and motivating problems. If the student is very absorbed and happy, then AutoTutor should be minimally directive.

At this point in the project, most of the emotion-sensing technologies have been assembled with AutoTutor, and the components, features, and representations of each of the sensing technologies are undergoing analysis. The channels currently being analyzed are as follows:

1) AutoTutor's log file with speech acts of student and tutor turns, as well as knowledge states achieved from the tutorial dialogue;
2) an upper facial sensor device developed by Roz Picard's Affective Computing Lab at the Massachusetts Institute of Technology (MIT), Cambridge [25], [26];
3) a body posture pressure measurement system;
4) a haptic pressure sensor for the mouse that was developed at MIT;
5) a keyboard pressure sensor.

Affect states will be classified on the basis of these five input channels. Computational models are being explored to perform these emotion analyses [25], [27].

V. Conclusion

In closing, natural language dialogue facilities are not expected to do a reasonable job in all conversational contexts. The quality of the discourse depends on the subject matter, the knowledge of the learner, the expected depth of comprehension, and the expected sophistication of the dialogue strategies. Natural-language dialogue facilities are unlikely to be impressive when the subject matter requires mathematical or analytical precision and when the user would like to converse with a witty or humorous conversation partner. In contrast, a natural-language dialogue facility is more feasible in applications that involve imprecise verbal content, low-to-medium user knowledge about a topic, and earnest literal replies. AutoTutor works well when tutoring students on qualitative domains and when the shared knowledge between the tutor and learner is low or moderate (rather than high). If the shared knowledge is high, then both dialogue participants (i.e., the computer tutor and the learner) will be expecting a higher level of precision of mutual understanding and, therefore, will have a higher risk of failing to meet each other's expectations. Therefore, tutoring is an ideal testbed for the research and development of natural-language dialogue systems. And of course, the practical benefit is helping children and adults learn.

REFERENCES

[1] A. C. Graesser, S. Lu, G. T. Jackson, H. Mitchell, M. Ventura, A. Olney, and M. M. Louwerse, "AutoTutor: A tutor with dialogue in natural language," *Behav. Res. Methods, Instrum., Comput.*, vol. 36, pp. 180–193, 2004.

[2] A. C. Graesser, N. K. Person, and D. Harter, "Teaching tactics and dialogue in AutoTutor," *Int. J. Artif. Intell. Educ.*, vol. 12, pp. 257–279, 2001.

[3] A. C. Graesser, K. VanLehn, C. Rose, P. Jordan, and D. Harter, "Intelligent tutoring systems with conversational dialogue," *AI Mag.*, vol. 22, pp. 39–51, 2001.

[4] V. Aleven and K. R. Koedinger, "An effective metacognitive strategy: Learning by doing and explaining with a computer-based cognitive tutor," *Cogn. Sci.*, vol. 26, pp. 147–179, 2002.

[5] K. Van Lehn, R. M. Jones, and M. T. H. Chi, "A model of the self-explanation effect," *J. Learning Sci.*, vol. 2, pp. 1–60, 1992.

[6] J. R. Anderson, A. T. Corbett, K. R. Koedinger, and R. Pelletier, "Cognitive tutors: Lessons learned," *J. Learning Sci.*, vol. 4, pp. 167–207, 1995.

[7] M. T. H. Chi, S. A. Siler, H. Jeong, T. Yamauchi, and R. G. Hausmann, "Learning from human tutoring," *Cogn. Sci.*, vol. 25, pp. 471–533, 2001.

[8] A. C. Graesser, N. K. Person, and J. P. Magliano, "Collaborative dialogue patterns in naturalistic one-on-one tutoring," *Appl. Cogn. Psych.*, vol. 9, pp. 359–387, 1995.

[9] A. C. Graesser, X. Hu, and D. S. McNamara, "Computerized learning environments that incorporate research in discourse psychology, cognitive science, and computational linguistics," in *Experimental Cognitive Psychology and Its Applications: Festschrift in Honor of Lyle Bourne, Walter Kintsch, and Thomas Landauer*, A. F. Healy, Ed. Washington, DC: Amer. Psychological Assn, to be published.

[10] T. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse Processes*, vol. 25, pp. 259–284, 1998.

[11] *How People Learn: Brain, Mind, Experience, and School*, J. D. Bransford, A. L. Brown, and R. R. Cocking, Eds., National Academy Press, Washington, DC, 2000.

[12] D. Sleeman and J. S. Brown, *Intelligent Tutoring Systems*. Orlando, FL: Academic, 1982.

[13] A. Olney, M. M. Louwerse, E. C. Mathews, J. Marineau, H. H. Mitchell, and A. C. Graesser, "Utterance classification in AutoTutor," in *Building Educational Applications Using Natural Language Processing: Proc. Human Language Technology—North American Chapter Assn. Computational Linguistics Conf. 2003 Workshop*, 2003, pp. 1–8.

[14] N. Person and A. C. Graesser, "Human or computer? AutoTutor in a bystander Turing test," in *Proc. 6th Int. Conf. Intelligent Tutoring Systems*, 2002, pp. 821–830.

[15] A. C. Graesser *et al.*, "Why/AutoTutor: A test of learning gains from a physics tutor with natural language dialogue," in *Proc. 25th Annu. Conf. Cognitive Science Soc.*, 2003, pp. 1–6.

[16] P. A. Cohen, J. A. Kulik, and C. C. Kulik, "Educational outcomes of tutoring: A meta-analysis of findings," *Amer. Educ. Res. J.*, vol. 19, pp. 237–248, 1982.

[17] P. Chipman, A. Olney, and A. C. Graesser, "The AutoTutor 3 architecture: A software architecture for an expandable, high-availability ITS," in *Proc. 1st Int. Conf. Web Information Systems Technologies (WEBIST 2005)*, 2005, pp. 466–473.

[18] A. T. Corbett, "Cognitive computer tutors: Solving the two-sigma problem," in *Proc. 8th Annu. Conf. User Modeling*, 2001, pp. 137–147.

[19] W. Xu and A. Rudnicky, "Task-based dialog management using an agenda," in *Proc. Applied Natural Language Processing/North American Association for Computational Linguistics (ANLP/NAACL) 2000 Workshop Conversational SystemsApplied Natural Language Processing and North American Association for Computational Linguistics*, May 2000, pp. 42–47.

[20] P. Tapanainen and T. Järvinen, "A nonprojective dependency parser," in *Proc. 5th Conf. Applied Natural Language Processing*, 1997, pp. 64–67.

[21] Microsoft .NET Technology Overview. Microsoft Corp., Seattle, WA. [Online]. Available: http://msdn.microsoft.com/netframework/technologyinfo/overview/default.aspx

[22] B. Ritchie. (2004) NET Languages. [Online]. Available: http://www12.brinkster.com/brianr/languages.aspx

[23] W. L. Johnson, J. W. Rickel, and J. C. Lester, "Animated pedagogical agents: Face-to-face interaction in interactive learning environments," *Int. J. Artif. Intell. Educ.*, vol. 11, pp. 47–78, 2000.

[24] J. Gratch, J. Rickel, E. Andre, J. Cassell, E. Petajan, and N. Badler, "Creating interactive virtual humans: Some assembly required," *IEEE Intell. Syst.*, vol. 17, pp. 54–63, 2002.

[25] R. W. Picard, *Affective Computing*. Cambridge, MA: MIT Press, 1997.

[26] A. Kapoor and R. Picard, "Real-time, fully automated upper facial feature tracking," in *Proc. 5th Int. Conf. Automated Face Gesture Recognition*, 2002, pp. 10–16.

[27] B. Kort, R. Reilly, and R. Picard, "An affective model of interplay between emotions and learning: Reengineering educational pedagogy—Building a learning companion," in *Proc. Int. Conf. Advanced Learning Technologies*, 2002, pp. 43–48.

Arthur C. Graesser (F'01) received the Ph.D. degree in psychology from the University of California at San Diego in 1977.

He is currently a Full Professor in the Department of Psychology and an Adjunct Professor in mathematical sciences at the University of Memphis, Memphis, TN. He is currently a Co-Director of the Institute for Intelligent Systems and Director of the Center for Applied Psychological Research. He was a Visiting Researcher at Yale University, New Haven, CT, in 1983, Stanford University, Stanford, CA, in 1984, and Carnegie Mellon University, Pittsburgh, PA, in 1991. His primary research interests are in cognitive science and discourse processing. More specific interests include knowledge representation, question asking and answering, tutoring, text comprehension, inference generation, conversation, reading, education, memory, expert systems, artificial intelligence, and human–computer interaction. He is currently Editor of the journal *Discourse Processes*. In addition to publishing nearly 300 articles in journals, books, and conference proceedings, he has written two books and has edited eight books, the most recent one being the *Handbook of Discourse Processes* (Mahwah, NJ: Lawrence Erlbaum Associates, 2003).

Patrick Chipman is currently working toward the Master's degree in cognitive psychology at the University of Memphis, Memphis, TN.

He is one of the developers of AutoTutor, particularly its overall system architecture, and has research interests in virtual environments and social cognition.

Brian C. Haynes received the B.S. degree in computer science from the University of Memphis, Memphis, TN, in 2004.

He was one of the developers of AutoTutor, particularly its 3-D simulations system, and has research interests in computer graphics and network applications.

Andrew Olney received the B.A. degree in linguistics with cognitive science from the University of London, U.K., in 1998 and the M.Sc. degree in evolutionary and adaptive systems from the University of Sussex, U.K., in 2001. He is currently working towards the Ph.D. degree in mathematical sciences at the University of Memphis, Memphis, TN.

He is one the developers of the AutoTutor system, particularly its natural language dialogue systems, and has research interests in computational linguistics and robotics.