Artificial Intelligence in Education
J.D. Moore et al. (Eds.)
IOS Press, 2001

Interactive Conceptual Tutoring in Atlas-Andes

Carolyn P. Rosé, Pamela Jordan, Michael Ringenberg, Stephanie Siler, Kurt VanLehn, Anders Weinstein
LRDC, University of Pittsburgh,
Pittsburgh, PA 15260, 412-422-0284,
rosecp@pitt.edu

Abstract. The goal of the Atlas project is to increase the opportunities for students to construct their own knowledge by conversing (in typed form) with a natural language-based ITS. In this paper we present the results of a comparative evaluation between a model tracing tutor, the Andes system [9], with the otherwise equivalent dialogue enhanced Atlas-Andes [6]. Andes is a model tracing tutor (MTT) that presents quantitative physics problems to students. The focus of Andes is to help students develop good physics problem solving skills. While Andes has been successful at this task, nevertheless, there is ample evidence to suggest that teaching students to solve physics problems is not all that is required to provide them with a solid grounding in physics. While students in elementary mechanics courses have demonstrated an ability to master the skills required to solve quantitative physics problems, a number of studies have revealed that the same students perform very poorly when faced with qualitative physics problems [13, 12, 11]. Atlas provides Andes with the capability of leading students through directed lines of reasoning that teach basic physics conceptual knowledge, such as Newton's Laws. The purpose of these directed lines of reasoning is to provide a solid foundation in conceptual physics to promote meaningful learning and to enable students to develop meaningful problem solving strategies. In this study students using the dialogue enhanced version performed significantly better on a conceptual post-test than students using the standard version of Andes.

1 Introduction

The goal of the Atlas project is to increase the opportunities for students to construct their own knowledge by conversing (in typed form) with a natural language-based ITS. Our previous research [6] [7] [16] [15] has produced reusable components and tools for facilitating the development of domain specific tutorial dialogue systems. The domain independent Atlas system [6] provides a general purpose planning engine and robust input understanding component that can be used to augment any tutoring system with dialogue capabilities.

Natural language dialogue, which involves language understanding, planning, and language generation, offers a number of attractive features for intelligent tutoring systems. First, providing the opportunity for natural language input allows the system to assess student understanding based on the direct evidence of the content of student explanations rather than based on the indirect evidence of the problem solving mistakes the students make. Natural language dialogue provides a context in which the system can tailor its presentation of material more directly to the students' needs, for example by addressing student misconceptions immediately as they arise in conversation. Secondly, it gives students the opportunity to gain experience using the language of the domain they are learning. In contrast to using short menu interfaces, requiring students to type an answer triggers recall memory as opposed to recognition memory. Natural language dialogue makes it possible to build a more sophisticated type of tutor. In addition to the hints that many current tutors give, we can extend the tutor's repertoire to include the types of remediation subdialogues seen in natural human tutorial dialogue.

In this paper we substantiate these claims by presenting the results of a comparative evaluation between a model tracing tutor, the Andes system [9], with the otherwise equivalent dialogue enhanced Atlas-Andes [6]. Atlas provides Andes with the capability of leading students through directed lines of reasoning, called Knowledge Construction Dialogues (KCDs), that teach basic physics conceptual knowledge, such as Newton's Laws. The purpose of these directed lines of reasoning is to provide a solid foundation in conceptual physics to promote deep learning and to enable students to develop meaningful problem solving strategies. In this study students using the dialogue enhanced version performed significantly better on a conceptual post-test than students using the standard version.

2 Motivation

Andes [9] is a model tracing tutor (MTT) that presents quantitative physics problems to students. The focus of Andes is to help students acquire good physics problem solving skills. It encourages students to learn actively by requiring them to derive most of the solution to the presented problems themselves. It facilitates transfer across media by making its graphical user interface (GUI) as much like a piece of paper as possible. It keeps students on track by providing immediate feedback. In particular, each problem solving action entered by students is highlighted either red or green to indicate whether it was correct or not. This basic feedback is termed flag feedback. Furthermore, when students get stuck in the midst of problem solving and request help, Andes provides hint sequences designed to help them achieve the goal of solving the problem as quickly as possible. Nevertheless, there is ample evidence to suggest that teaching students to be good problem solvers is not enough to provide them with a solid grounding in physics.

While students in elementary mechanics courses have demonstrated an ability to master the skills required to solve quantitative physics problems, a number of studies have revealed that the same students perform very poorly when faced with qualitative physics problems [13, 12, 11]. Furthermore, the naive conceptions of physics that they bring with them when they begin a formal study of physics do not change significantly by the time they finish their classes [13]. Similarly, MTTs in a wide range of domains have commonly been criticized for failing to encourage deep learning [18]. If students do not reflect upon the hints they are given, but instead simply continue guessing until they perform an action that receives positive feedback, they tend to learn the right actions for the wrong reasons [2, 3].

This shallow learning problem was initially addressed in the Andes tutoring system with the development of the Conceptual Helper [1]. In contrast to the basic Andes hint sequences that are available upon request, the Conceptual Helper offered students unsolicited help as soon as the student showed evidence of needing that help. Help was provided in the form of mini-lessons involving text and graphics that provided students with detailed lines of reasoning designed to promote a conceptual understanding of physics. An experimental group

Tutor1: Let's think about the difference between speed and velocity. A closely related distinction is that of the difference between distance traveled and displacement from the origin. Take as an example a bumblebee flying from point A to point B by means of a curvy path. If you draw a vector from point A to point B, you will have drawn the bee's displacement vector. What does the displacement vector represent?

Student1: The distance travelled by the bee from its starting point to its ending point.

Tutor2: The displacement vector only represents the net change in position accomplished by the bee's motion. So the displacement vector does not tell you how far the bee traveled. It only tells you the net change in the bee's position. If you want to know the actual distance traveled by the bee, what entity do you have to measure the length of?

Student2: The bee's path.

Tutor3: The bee's displacement vector is obviously a vector. Is the actual distance traveled by the bee a vector or a scalar?

Student3: Scalar.

Tutor4: So distance traveled is the scalar counterpart to displacement, which is a vector quantity. Now, can you tell me what is the difference between speed and velocity?

Student4: Velocity is a vector composed of speed and direction.

Tutor5: Speed is a scalar. So when you measure the bee's speed, do you want to use the distance traveled, which is a scalar, or the displacement, which is a vector?

Student5: Distance travelled.

Tutor6: So the equation for speed is the length of the path traveled by the body divided by duration of the motion, regardless of how curvy the path is, even if the path leads back to where the body started from.

Figure 1: **Real example interaction between student and Atlas**

using Andes with the added Conceptual help performed significantly better than a Control group using only the standard Andes hint sequences and flag feedback. The text and graphics provided by the Conceptual Helper allowed students to learn by means of receptive learning, which has been demonstrated to be successful when students have both the necessary prior knowledge to integrate the desired new knowledge in a meaningful way and sufficient motivation to learn meaningfully rather than in a shallow or rote manner [4].

The difference between the type of help offered via the Conceptual Helper and the type of help offered by Atlas is that Atlas provides interactive directed lines of reasoning, such as that displayed in Figure 1. In the Atlas directed lines of reasoning, students are led step by step by a sequence of questions designed to guide them into discovering conceptual knowledge as independently as possible. Interactive directed lines of reasoning provide advantages over the receptive learning approach. Firstly, students are required to answer questions at every step in the directed line of reasoning. Thus, when their answer reveals a lack of required prior knowledge, it can be addressed in order to allow the directed line of reasoning to proceed meaningfully. Therefore, the dialogue is tailored to the individual differences in prior knowledge among students. Secondly, requiring students to answer questions encourages students to participate in an active way.

3 Integrating Atlas and Andes

In a prototype version of Atlas-Andes [6], interactive directed lines of reasoning were linked into Andes at the end of the standard Andes hint sequences. In order to increase exposure to conceptual tutoring, however, in the current version of Atlas-Andes interactive directed lines of reasoning are linked into Andes via the Conceptual Helper, instead of the original non-interactive mini-lessons.

The Conceptual Helper makes use of the Andes solution graph corresponding to the current problem for diagnosing student errors. Each time the student performs a GUI action that is determined to be incorrect, the Conceptual Helper attempts to match student actions to the closest action in the solution graph that has not been successfully accomplished yet. Thus, it attempts to determine which correct action the student was most likely trying to accomplish. It then checks to see if any rule higher up in the solution graph that derives that step has a KCD associated with it. If it identifies one or more such KCDs that the student has not seen since the beginning of the student's current session with Andes, then Andes requests Atlas to launch the KCD from that set that is closest to the matching step in the solution graph. Control is then passed from Andes to Atlas, and students are prevented from performing any of the regular Andes GUI actions until the student either finishes the KCD or dismisses it.

In contrast to the more typical Andes help in the form of hints available on demand, when the Conceptual Helper finds a potential match between an incorrect action and a step in the solution graph, it provides unsolicited help to the student on the rule that derives that step. Thus, by using the Conceptual Helper to initiate KCDs, directed lines of reasoning are provided in an unsolicited fashion when the students first show evidence of needing such tutoring by performing buggy problem-solving actions. Unfortunately, this approach to integrating Atlas and Andes prevents Andes from selecting KCDs in the full range of instances in which it would be advantageous to do so. For example, since equation errors are only diagnosed in Andes on demand, equation errors are never identified by the Conceptual Helper. Similarly, since the Conceptual Helper works by matching incorrect actions to steps in the Andes solution graph, it is not able to identify problems of omission where a student simply does not know what to do next.

4 Atlas Dialogue System Components

The two main components of Atlas are APE [7], the Atlas Planning Engine, and CARMEL [16], the natural language understanding (NLU) component.

Planning is required in dialogue-based tutoring systems to ensure that the dialogues that are produced sound coherent and that the tutor's pedagogical goals are accomplished. APE is a "just in time" planner specialized for easy construction and rapid generation of hierarchically organized dialogues. APE is a reactive planner [8, 19]. Thus, it has the ability to perform micro-adaptive modifications to its agenda in order to better adapt to the student's needs. For example, if one considers that a strategy may be encoded as multiple steps in a single operator, the remainder of a strategy can be skipped if the student's response to a question at a certain step within that strategy indicates that it is not an appropriate strategy for that student. At that point it can either replace the current strategy with a different one that has a matching goal, or it can remove or replace one or more goals from the top of the agenda. Another way in which APE is reactive is by making its decisions for how to respond to a student contingent upon the agenda's goal hierarchy at the point when it considers alternative decompositions for achieving the current goal.

The task of Atlas's input understander is to extract relevant information from student answers and other natural language input to pass back to the planner. CARMEL provides a broad foundation for language understanding. It is composed of a broad coverage English syntactic parsing grammar and lexicon; robust and efficient algorithms for parsing, semantic interpretation, and repair; and a formalism for entering idiomatic and domain specific semantic knowledge. The goal behind the CARMEL approach is to achieve the most complete deep analysis possible within practical limits by relaxing constraints only as needed. CARMEL first attempts to construct analyses that satisfy both semantic and syntactic well-formedness constraints. A spelling corrector [5] is integrated with the lexical look-up mechanism in order to robustly recognize the student's intended input even in the face of typos and spelling errors. The robust parser [17] has the ability to efficiently relax syntactic constraints as needed and as allowed by its parameterized flexibility settings. For sentences remaining beyond the coverage of its syntactic knowledge, a repair stage [16], relying solely on semantic constraints compiled from a meaning representation specification, is used to assemble the pieces of a fragmentary parse. Thus, robustness techniques are applied progressively, on an as-needed basis in order to efficiently address the wide range of phenomena that make language understanding challenging.

APE and CARMEL are both completely domain independent and reusable. Nevertheless, a significant knowledge engineering effort is required to develop the knowledge sources required to adapt them for use within a specific domain. In particular, an operator library is required for APE and a meaning representation specification is required for CARMEL. Atlas-Andes currently provides fifty-five separate directed lines of reasoning that teach the basic conceptual rules targeting all aspects of Newtonian Mechanics. Authoring the knowledge sources for all of these KCDs would be a formidable task. Thus, to expedite the process of constructing these knowledge sources, we have developed the KCD Authoring Tool Suite [14, 15].

The KCD Authoring Tool Suite provides a GUI interface for authoring knowledge construction dialogues at a high level. The knowledge sources for APE are compiled automatically from the specification authored through a GUI KCD editor. Some user interaction is required in the process of generating the language understanding knowledge sources from the specification. The knowledge sources required for the complete set of KCDs currently provided by Atlas-Andes required only three man months of authoring time, including initial authoring, review, revision, and pilot testing. Only 3 days of this time were devoted specifically to working on the language understanding knowledge sources. Thus, the majority of the time was dedicated to designing effective pedagogy, which is where we believe the majority of time is best spent. It should be noted that in the initial version of the KCD Authoring Tool Suite we do not make full use of the power provided either by APE or by CARMEL. In particular, the current version of Atlas uses APE as a finite-state push down automaton, which is less powerful than a reactive planner although it is a very common mechanism used in dialogue systems for tutoring and other applications. For language understanding the

KCD Authoring Tool Suite bootstraps simple semantic grammar rules for CARMEL rather than building semantics on top of the deep syntactic functional analysis provided by the CARMEL grammar. Thus, the approach to language understanding taken here is similar to the information extraction approach taken in CIRCSIM-TUTOR [10]. The one feature provided by CARMEL that we make use of that was not part of Glass's approach is the ability to skip words from the student's input text in order to match an extraction pattern from the semantic grammar.

The design of each of Atlas-Andes's fifty-five directed lines of reasoning began with the development of a main line of reasoning consisting of a sequence of tutorial goals, each realized as a question possibly accompanied by a short explanation. When the main line of reasoning is authored, each question is associated with one or more expected answers. For example, the question, "Is tension force in a string oriented towards or away from a body that is suspended from the string?" has two expected answers, namely "towards" and "away". The correct answer is "away". Every question also has a catch-all `anything else` case, so that every possible student input can be covered. Together, the expected answers and `anything else` constitute a set of answer categories. When the question is presented to the student, and the student responds with some arbitrary text, the NLU component assigns the student's text to one of these answer categories based on its semantic grammar, which defines the range of ways concepts may be encoded in student input texts.

A set of one or more remediation goals is associated with each expected wrong or partial student answer as well as the `anything else` case. Each remediation goal has one or more corresponding lines of reasoning, each consisting of a sequence of tutorial goals as in the main line of reasoning. Remediation directed lines of reasoning take on a number of different forms. In simple cases, the student is simply presented with a short explanation to clear up a detected misconception or missing piece of information. In other cases, a more specific or simpler version of the previous question is given in an attempt to draw the correct answer out of the student. In cases where the missing or faulty knowledge can be decomposed into a sequence of more basic pieces of knowledge, a multi-step directed line of reasoning is used. Thus, when a student answers a question incorrectly, a line of reasoning corresponding to the associated remediation goal is initiated. When that subdialogue is completed, the main line of reasoning proceeds. Thus, the directed lines of reasoning provided by Atlas-Andes take on a recursive, hierarchical structure, allowing a great deal of flexibility in adapting to specific student needs.

The remediations for the `anything else` cases are designed with the intent of sounding natural almost no matter what the student types. For example, in some simple cases the correct answer to the question is simply stated. If the student's answer was indeed a correct way of articulating this answer, then in stating the correct answer, the system is meant to come across as simply confirming the correct answer in order to transition into the next point. If the student's answer was not correct, it is meant to come across as correcting the student. In other cases, the remediation subdialogue for the `anything else` case begins by asking a related question. For example, as part of the KCD for teaching the direction of tension forces, Atlas asks what force it is that keeps a body that is attached to a taught, stationary string from moving. An example answer from our corpus that was correctly classified as `anything else` is "the string". The remediation subdialogue begins with, "In order for the body to not move downward, an upward force must balance the downward force of its weight. What force is that?" Here a hint is given, and the original question is rephrased. But even if the student had answered correctly, this tutor turn could be seen as a coherent continuation of the dialogue since it could be seen as a slightly different question than the original one. In this case, the student's answer is incorrect but related to the correct answer. So the hint plus question combination comes across as directing the student to be more precise. While this approach to maintaining coherency in the face of possible language understanding errors is not fool proof, in practice we have found that it is rare for the NLU component to incorrectly classify a correct answer as `anything else`. For example, in the evaluation reported in Section 6, there were only 12 instances out of 355 student input texts.

5 Knowledge Construction Dialogue Design in Atlas-Andes

On a high level, the Atlas-Andes KCDs were all designed to follow the same basic format. They begin with an introduction to the main tutorial objective of the KCD as well as the scenario that will be used to illustrate that concept. The scenario is then used as a tool over a series of exchanges with the student to help draw out the desired conceptual knowledge by providing the student with opportunities to make observations and predictions. Finally, the tutor summarizes the main point in the final turn.

Figure 1 contains a full example knowledge construction dialogue taken from our pilot testing corpus. Thus, it is an actual interaction between a student and the system. A detailed look at this example KCD illustrates the design principles behind the development of the complete set of Atlas-Andes KCDs. This particular KCD was designed to lead the student into an understanding of the differences between speed and velocity, first in terms of what they are, and then in terms of how they should be computed. As is typical in the Atlas-Andes directed lines of reasoning, an every day scenario is used to illustrate the concept, in this case a bee flying. Familiar scenarios are used in order to make the directed lines of reasoning accessible to the broadest possible audience.

To introduce the student to the main tutorial objective of the KCD, in the first tutor turn the student is presented with an *advance organizer* [4]. This brief overview sets up the student's expectation for what the tutor's argument will be and to bridge the gap between what the student may already know and what the student should learn, in this case the distinction between speed and velocity. In the same turn, the tutor introduces the scenario that will be used to point the student towards the desired conclusions, in this case the flight of a bee. The turn ends with a question about what is represented by a displacement vector.

The student responds in the first student turn by giving a common wrong answer, namely that displacement is the distance traveled by the bee. The distinction between distance and displacement was introduced in the advance organizer as a relevant piece of possible prior knowledge. This example dialogue illustrates how Atlas-Andes handles the case where the relevant prior knowledge is missing. The tutor begins to address the missing knowledge in the second tutor turn by briefly explaining the distinction between distance and displacement. A check question is then given to the student to ensure that the student has properly assimilated the explanation. The student responds with a correct answer, indicating that the tutor explanation was successfully integrated into the student's understanding.

Thus, the advance organizer is used to draw the student's attention to possible prior knowledge that is relevant to the current directed line of reasoning. But if that knowledge turns out to be missing, the tutor can address the student's deficit before proceeding. In this case, the advance organizer still serves the purpose of allowing the student to understand the relevance of the inserted remediation directed line of reasoning.

When the remediation subdialogue is complete, the tutor continues presenting the student with questions, which the student is able to answer almost all correctly. The student's answer to the question about the difference between speed and velocity is slightly off, so the system includes a clarification in the beginning the subsequent turn, namely that speed is a scalar quantity, since this is important for the explanation that follows. The student arrives at the desired conclusion in the final student turn. As usual, the tutor then responds with a brief summary of the argument.

6 Evaluation of Atlas-Andes

To assess the effectiveness of the current version of Atlas-Andes, we ran a small comparative evaluation of Andes without dialogue capabilities with Atlas-Andes. The results demonstrated a significant effect in favor of Atlas-Andes.

Participants The participants were recruited from three first-year physics classes at the University of Pittsburgh. With the instructors' permissions, at the beginning of the physics classes we briefly described the evaluation to the class. Twenty-eight students from these classes took the conceptual pretest. These students were then divided into two sets with roughly the same average pretest score. The students were then given the option of attending a half-hour Andes tutorial in which they were walked through a simple Andes problem step by step. Next, the students were assigned eight Andes homework problems. The students in the Atlas-Andes condition used the dialogue enhanced Atlas-Andes system. The students in the Andes condition solved the problems using Andes without dialogue capabilities. To ensure that the Andes only students had the opportunity to learn the same material presented to the Atlas-Andes students, the Andes only students were provided with short conceptual mini-lessons in place of KCDs that covered the same conceptual knowledge but were not interactive and were not illustrated with the scenarios used in the KCDs. Of the 28 students who signed up for the study, 23 came to the computer lab at least once to work on the homework problems. Only twelve students completed all eight homework problems. The most common reason cited for dropping out of the study was insufficient time given the demands of class obligations. At the end of a two week period, the students took a conceptual post-test and a problem solving post-test.

Attrition statistics The mean pretest scores for students who did and did not finish all eight Andes problems were 24.5 and 19.4, respectively, a marginally significant difference, $t(26) = 1.77$, $p < .10$. Thus, the students who completed homework using Andes had more prior conceptual physics knowledge than students who dropped out. However, for our purposes of comparing student learning in the Andes condition to learning in the Atlas-Andes condition, we were interested in differences between conditions that may contribute to outcome measures. There was no significant difference between conditions (Andes and Atlas-Andes) in the likelihood of dropping out of the study, c2 (1, N=28) = 1.2, $p = .27$. Nor was there a difference between conditions in pretest scores of the twelve students who completed all homework (24 and 25.2, respectively), $t(10) = 0.25$, $p = .81$.

Materials The pretest consisted of eight problem solving questions, similar to those in Andes. These problems were each broken into steps that each asked students to draw something (e.g., axes, a vector, etc.), give a formula, or make a specific computation. The Andes problems assigned as homework during the evaluation were similarly broken down into steps. The conceptual post-test consisted of 34 multiple-choice conceptual physics questions. Each question on the conceptual post-test was roughly equivalent in scope to a step on the conceptual pretest. Each of these questions was designed to test knowledge of a single physics concept. The complete set of principles taught by the KCDs was a subset of the principles tested on the conceptual post-test. The problem solving post-test consisted of three problem solving questions not broken down into steps.

Results Students who completed all homework problems were paired with a student in the other condition who had a similar pretest score (within 5 points) and the same physics instructor. There was no difference between conditions on scores on the conceptual pretest for the students who were paired (unpaired, two-tailed t-test: $t(8) = 0.43$, $p = .68$). Using these paired students only (10 of 12 who had finished all homework problems), students in the Atlas-Andes condition scored significantly higher than students in the Andes condition on the conceptual post-test, (paired two-tailed t-test: $t(8) = 2.32$, $p < .05$), with an effect size of .9 standard deviations. Because so few participants finished the study, we wanted to more directly asses the effect of receiving KCDs to support the overall significant results. The conceptual post-test questions were divided into those addressing concepts/rules at least one of the students in the Atlas-Andes condition had received a KCD for and those addressing concepts for which none of the students in the Atlas-Andes condition had received a corresponding KCD. On questions addressing concepts for which at least one of the students in the Atlas-Andes condition received a KCD, there was a highly significant difference between groups, $t(8) = 5.84$, $p < .001$. This difference could not be explained by higher pretest scores in the Atlas-Andes condition on pretest concepts later addressed by KCDs (M [Andes] = 4.2; M [Atlas-Andes] = 4.8), $t(8) = 0.30$, $p = .78$. However, for post-test questions for which none of the students in the Atlas-Andes condition received a KCD, the difference between conditions was not significant, $t(8) = 1.45$, $p = .19$. Again, there were no pretest differences for this subset of the pretest (M [Andes] = 18.8; M [Atlas-Andes] = 20.4), $t(8) = 0.31$, $p = .77$. The results on the problem solving post-test showed no significant difference between groups.

Discussion On average, students in the Atlas-Andes condition only saw four KCDs in the course of solving their eight Andes problems. Thus, only a small fraction of their total time with Andes was spent participating in dialogues with the system. Nevertheless, with two exceptions, students always answered conceptual post-test questions correctly when they pertained to KCDs they saw. The only exceptions were two students who each participated in a dialogue about the direction of dynamic friction force but still did not draw a friction force vector in one post-test question. These results demonstrate both the practicality and the effectiveness of tutorial dialogue for teaching a conceptual understanding of physics.

The simplistic approach to dialogue management and language understanding used in this evaluation was surprisingly successful. In an evaluation of how often the NLU component was able to correctly classify the students' input texts into the authored answer categories, we found that it was correct 96.3% of the time, with a precision of 99.5% and a recall of 94.4%. Since only 62.3% of student answer texts received a classification other than `anything else`, we conducted an additional evaluation of the provided answer classes to determine whether our high level of performance on student answer handling was due to including too few answer classes in the authored KCDs, thus making the task simpler than it should have been. To this end, the log files were evaluated by a retired physics professor not involved in the original authoring of the KCDs to determine how often the student texts that were correctly classified as `anything else` according to the authored answer categories could have better been handled by including an additional special case answer class with its own specific remediation designed for it. it was determined that only 7 out of the 122 correctly assigned `anything else` cases could better be handled this way. Over the entire set of 355 student input texts, this only constitutes 2% of student answers. Thus, 98% of the time, the authored answer classes were sufficient for handling student input.

7 Current Directions

In this paper we have described the current version of Atlas-Andes, a dialogue based tutoring system integrating the Andes physics tutoring system with the Atlas tutorial dialogue system. Our comparative evaluation between Andes and the dialogue enhanced Atlas-Andes showed a significant difference in learning gains in favor of Atlas-Andes on a conceptual post-test. With the help of the KCD Authoring Tool Suite, development of the dialogue capabilities for Atlas-Andes required only 3 months of development time. These results demonstrate both that it is currently practical to build dialogue enhanced tutoring systems and that students learn more for the dialogue enhanced version than from a standard model tracing tutor alone.

Nevertheless, we believe there is much room for further exploration and development. Five issues dominate Atlas' future development. First, writing an effective KCD and debugging it with real students is an inherently labor-intensive process. We will continue building tools to expedite the process. Secondly, the conventional wisdom is that a recursive finite state network does not provide sufficient flexibility for managing complex dialogs. Although Atlas' networks are interpreted by a reactive planner, we do not currently make use of all of its power. Thus a second important direction is to determine how much of this additional power is necessary for conducting more effective tutorial dialogues with students. Similarly, the current version of Atlas does not make use of the full power offered by the CARMEL core understanding component. Thus, another related direction is determining how sophisticated of an analysis of student input is necessary for the system to determine how best to proceed with its dialogue with the student. Fourthly, we have deliberately left the Andes system's two major knowledge sources alone, so Andes is still responsible for managing the physics problem solving process and for deciding which hint sequence is appropriate. This implies that KCDs are used mostly to replace hint sequences. We are not sure if this simple design will allow pedagogically useful dialogs, or whether we will need to port some of Andes' knowledge to Atlas. Finally, we plan to extend Atlas' capabilities to additional types of knowledge construction dialogues, such as goal scaffolding dialogues.

8 Acknowledgements

This research was supported by the Office of Naval Research, Cognitive Science Division under grant number N00014-0-1-0600 and by NSF grant number 9720359 to CIRCLE, the Center for Interdisciplinary Research on Constructive Learning Environments at the University of Pittsburgh and Carnegie-Mellon University.

Special thanks to all of the members of the Andes research group for providing us with such a rich and exciting environment in which to develop this work!

References

[1] P. L. Albacete. *An Intelligent Tutoring System for Teaching Fundamental Physics Concepts.* PhD thesis, University of Pittsburgh, 1999.

[2] V. Aleven, K. R. Koedinger, and K. Cross. Tutoring answer-explanation fosters learning and understanding. In *Proceedings of Artificial Intelligence in Education*, 1999.

[3] V. Aleven, K. R. Koedinger, H. C. Sinclair, and J. Snyder. Combating shallow learning in a tutor for geometry problem solving. In *Proceedings of the Intelligent Tutoring Systems Conference*, 1998.

[4] D. Ausubel. *Educational Psychology: A Cognitive View.* Holt, Rinehart and Winston, Inc., 1978.

[5] M. Elmi and M. Evens. Spelling correction using context. In *Proceedings of COLING/ACL 98*, 1998.

[6] R. Freedman, C. P. Rosé, M. A. Ringenberg, and K. VanLehn. Its tools for natural language dialogue: A domain-independent parser and planner. In *Proceedings of the Intelligent Tutoring Systems Conference*, 2000.

[7] R. K. Freedman. Plan-based dialogue management in a physics tutor. In *Proceedings of the 6th Applied Natural Language Processing Conference*, 2000.

[8] M. P. Georgeff and F. F. Ingrand. Decision-making in an embedded reasoning system. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI '89)*, 1989.

[9] A. S. Gertner and K. VanLehn. Andes: A coached problem solving environment for physics. In *Proceedings of ITS 2000*, 2000.

[10] M. S. Glass. *Broadening Input Understanding in an Intelligent Tutoring System.* PhD thesis, Illinois Institute of Technology, 1999.

[11] R. R. Hake. Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics students. *American Journal of Physics*, 66(64), 1998.

[12] I. A. Halloun and D. Hestenes. Common sense knowledge about motion. *American Journal of Physics*, 53(11):1056–1065, 1985.

[13] I. A. Halloun and D. Hestenes. The initial knowledge state of college physics students. *American Journal of Physics*, 53(11):1043–1055, 1985.

[14] Pamela Jordan, Carolyn Rosé, and Kurt VanLehn. Tools for authoring tutorial dialogue knowledge. In *Proceedings of AI in Education 2001 Conference*, 2001.

[15] C. P. Rosé. Facilitating the rapid development of language understanding interfaces for tutoring systems. In *Working Notes of the AAAI Fall Symposium on Building Tutorial Dialogue Systems*, 2000.

[16] C. P. Rosé. A framework for robust semantic interpretation. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics*, 2000.

[17] C. P. Rosé and A. Lavie. Balancing robustness and efficiency in unification augmented context-free parsers for large practical applications. In J. C. Junqua and G. Van Noord, editors, *Robustness in Language and Speech Technologies*. Kluwer Academic Press, 2001.

[18] K. VanLehn, R. Freedman, P. Jordan, C. Murray, R. Osan, M. Ringenburg, C. P. Rosé, K. Schultze, R. Shelby, D. Treacy, A. Weinstein, and M. Wintersgill. Fading and deepening: The next steps for andes and other model-tracing tutors. In *Proceedings of the Intelligent Tutoring Systems Conference*, 2000.

[19] D. Wilkins, K. Myers, J. Lowrance, and L. Wesley. Planning and reacting in uncertain and dynamic environments. *Journal of Experimental and Theoretical Artificial Intelligence*, 7:972–978, 1995.