Fading and Deepening: The Next Steps for Andes and other Model-Tracing Tutors

Kurt VanLehn[1], Reva Freedman[1], Pamela Jordan[1], Charles Murray[1], Remus Osan[1], Michael Ringenberg[1], Carolyn Rosé[1], Kay Schulze[3], Robert Shelby[2], Donald Treacy[2], Anders Weinstein[1], and Mary Wintersgill[2]

Abstract. Model tracing tutors have been quite successful in teaching cognitive skills; however, they still are not as competent as expert human tutors. We propose two ways to improve model tracing tutors and in particular the Andes physics tutor. First, tutors should fade their scaffolding. Although most model tracing tutors have scaffolding that needs to be gradually removed (faded), Andes' scaffolding is already "faded," and that causes student modeling difficulties that adversely impact its tutoring. A proposed solution to this problem is presented. Second, tutors should integrate the knowledge they currently teach with other important knowledge in the task domain in order to promote deeper learning. Several types of deep learning are discussed, and it is argued that natural language processing is necessary for encouraging such learning. A new project, Atlas, is developing natural language based enhancements to model tracing tutors that are intended to encourage deeper learning.

[1] University of Pittsburgh, Learning Research and Development Center, Pittsburgh, PA, 15260 (vanlehn@cs.pitt.edu)
[2] United States Naval Academy, Department of Physics
[3] United States Naval Academy, Computer Science Department

1 Model Tracing Tutors

This paper considers how to improve an already successful type of intelligent tutoring system, the model-tracing tutor (MTT). Although Anderson, Boyle and Reiser (1985) coined the term, MTT refers to a relatively broad class of intelligent tutoring systems, namely those that contain a model of the cognition that one would like students to engage in and a means of encouraging students to reason in that fashion. An MTT usually has three modules: an expert model, a graphical user interface and a pedagogical module. After describing these modules below, we describe in the following sections the two main problems that MTT face and our proposed solutions to them. We conclude with a short comparison of the proposed MTT and human tutors.

An MTT contains an *expert model* which models how the designers would like students to reason. It has problem solving strategies that are coherent, precise, complete, and often quite simple. Sometimes the strategies are designed to enhance learning rather then accurately replicate expert strategies. They may require doing steps that students and even some instructors do not usually do, such as solving algebra word problems by writing arithmetic equations before writing algebraic ones (Aleven, Koedinger, Sinclair, & Snyder, 1998).

Most MTT have a *high bandwidth graphical user interface* (GUI). A GUI has high bandwidth if it has students display most of their reasoning, typically by requiring them to enter more information than they would if they were working as normal on a sheet of paper (VanLehn, 1988). For instance, students might have to define variables, as in the Andes physics tutor (Gertner & VanLehn, 2000), or provide coherent labels for columns in tables and axes in graphs (Aleven, Koedinger, & Cross, 1999; Aleven et al., 1998). Some GUI even have students maintain a goal tree (e.g., Koedinger & Anderson, 1993; Reiser, Beekelaar, Tyle, & Merrill, 1991; Singley, 1990). Many MTT designers claim that explicating such information increases learning, and there is some evidence for that claim (e.g., Merrill & Reiser, 1994; Singley, 1990).

An MTT also contains a *pedagogical module* that can provide immediate feedback and hints. Feedback is given whenever the student's action does not match the expert model's action. Help is usually provided via a hint sequence. A hint sequence starts with a general hint, then allows the student to try again. If the student's new action is also incorrect or the student ask for more help, then the tutor gives the next hint in the sequence, which provides more information about what the target action should be. The hints become more specific until the student enters the target action correctly. If the tutor runs out of hints, it either tells the student exactly what to enter or does the action for the student. Clearly, the pedagogical module is based on the hypothesis that immediate feedback and hint sequences facilitate learning. There have been several studies of this hypothesis (e.g., Anderson, Corbett, Koedinger, & Pelletier, 1995; Mark & Greer, 1995).

2 Fading

Although MTTs have an enviable track record (e.g., Anderson et al., 1995; Koedinger, Anderson, Hadley, & Mark, 1995; McKendree, Radlinski, & Atwood, 1992; Reiser, Copen, Ranney, Hamid, & Kimberg, in press; Shelby et al., in prep.), they are sometimes criticized as being too rigid. Below, we review such claims then discuss "fading," which is an obvious solution to the rigidity problems. However, the bulk of this section concerns some non-obvious problems that fading causes, how they emerged in the Andes tutoring system, and what we propose to do about them.

MTTs are sometimes criticized for allowing only one strategy for problem solving when many are possible. For instance, Anderson and Corbett's Lisp tutor has been criticized for forcing students to enter code top-down (e.g., Reiser, Kimberg, Lovett, & Ranney, 1992). The critics argue that when the tutor keeps students on the solution path of a single strategy, it prevents the students from inventing their own strategies, testing them and refining them.

A second complaint is that the tutor often forces students to enter information that they often try to hold in their working memory. This could prevent students from learning how to manage their use of memory. For instance, when solving complex algebra equations, some students try to write down fewer intermediate steps then expert mathematicians (Lewis, 1981). Such students need to learn to write more down and trust their memory less, but the rigid scaffolding of MTT GUIs thwarts such learning because it controls how much they must write down.

A third criticism is that the tutor provides too much scaffolding of error handling. Because the tutor detects errors for the student and hints at how to correct them, when students of MTTs are tested with the tutor absent, they are often worse at detecting and correcting their own errors than students who covered the same material without the tutor (Reiser et al., in press).

As Collins, Brown and Newman (1989), McArthur (1990) and others have noted, expert human tutors often start with large amounts of scaffolding, then *fade* it as students exhibit more competence. Similarly, MTTs should fade the procedural restrictions on problem solving, thus allowing students to solve problems any way they want, to experiment with strategies and to manage their use of memory. MTTs should also fade their support for error handling, thus allowing student to learn how to detect and correct errors by themselves.

2.1 Andes is Already Partially Faded

Although most MTTs should fade their scaffolding, our MTT, Andes (Gertner & VanLehn, 2000; VanLehn, 1996) has the opposite problem. It has turned out to have too little scaffolding. Although Andes flags incorrect entries by turning them red, it allows students to enter steps in any order and to omit almost any step. This gives Andes' students the freedom to discover strategies and manage their use of memory. On the other hand, the lack of procedural restrictions has caused 2 pedagogical problems.

First, some students failed to develop effective problem solving strategies, and Andes' hints about what step to do next only frustrated and confused them. The log files indicate that many Andes students do not follow a single strategy. They often mix steps from multiple strategies, probably without knowing that they are doing so. When a student asks for advice on what to do next, Andes uses probabilistic reasoning to guess the step the student might be trying to do next and construct a hint sequence leading up to it (Gertner, Conati, & VanLehn, 1998; Gertner, 1998). The students sometimes can't figure out why that step should be next. Indeed, there usually is no good explanation because their strategy wasn't coherent. Some students felt quite frustrated by the apparent randomness of Andes' advice, not realizing that it was caused by their own random behavior.

Second, when students received hints on errors, they often could not fix the errors themselves. Instead, they proceeded all the way to the last hint in the sequence, which would tell them exactly what to enter. Andes hint sequences are no different in design from those used successfully by other MTT, which suggests that it is the *context* of the hints which is causing them to fail. For instance, a common error was to omit a negative sign that was introduced by projecting a vector. Andes' first hint on a sign error is to "check your signs." If the student has drawn the appropriate vector and is working on projecting it (e.g., entering an equation such as $V_x=-V*\cos\theta$), then this hint would probably work fine. However, the students who make this error have often skipped both drawing the vector and writing the projection equation down. When they receive the "check signs" hint, they can't figure out where the negative sign should have come from because they did the calculations in their head. Many hints fail not because they are bad hints, but because Andes let student do so many critical steps in their head that the students cannot reconstruct their reasoning in order to find the error.

From these observations during formative evaluations, it has become clear that although Andes' faded scaffolding allows students to invent their own strategies, repair their own errors, and manage their use of memory, some students seem to need more scaffolding. They should be constrained to use a single strategy and to not skip key steps. That is, some students need the rigid, procedural scaffolding that most MTT have but Andes lacks.

2.2 Micro-Adaptive Fading

A straightforward approach to fading is to equip the MTT with different levels of scaffolding. Novice students start with a high level of scaffolding. As their competence increases, the tutor reduces (fades) the level of scaffolding. This is a *macro-adaptive* approach to fading (cf. Shute, 1993). The tutor changes its level of scaffolding in between problems, but during a problem's solution, the level remains the same. Inspired by our analysis of human tutors, we are revising Andes to use a *micro-adaptive* approach to fading. The scaffolding only occurs at the moment the student seems to need it.

The key idea is that although Andes will be able to recognize a wide variety of strategies, skipped steps, etc., it will only give advice and help on a single strategy. The strategy to (a) select a quantity whose value is sought, (b) decide which major physics principle is appropriate for finding it, (c) execute the procedure for that principle, (d) figure out which quantities still need to be found, then repeat the cycle from step (a). This strategy is well known in physics, and the procedures often appear in textbooks. Some tutoring systems (e.g., Reif & Scott, 1999) constrain students to follow this strategy. The physicists on the Andes project have designed the specific strategy to be taught by determining which physics principles are the major ones and designing procedures for them. For instance, Newton's second law (F=m*a) is a major principle, but the weight law (W=m*g) is not. Instead, it is included in the Newton's law procedure. The physicists also determined which steps in the procedures must be entered by student on the Andes GUI, and which can be done mentally if the student wishes.

The new Andes will not constrain students to follow the strategy. Just as before, students can enter steps in any order, skipping as many as they like. As long as they enter a correct step, it will turn green. Moreover, if they can fix their incorrect (red) steps without help, then they will see no signs of the target strategy. However, if they ask for help, then Andes will try to get them to follow the target strategy.

More specifically, if they are stuck and ask for a hint on what step to do next, Andes will start by asking them, "What quantities are you seeking?" and offering a menu. Incorrect menu selections evoke feedback and a hint sequence. When the student has selected a quantity that is actually sought in this problem, then Andes asks, "What principle should be used to find it?" and offers a menu. Again students get feedback and hints until they have chosen a principle appropriate for the problem and the sought quantity. Andes then matches the principle's procedure to the student's entries and determines the first step that the student has not yet done. It gives hints on that step. When the student finally does the target step, this help episode is over, and the student resumes solving the problem. If the student immediately asks for another hint on what to do next, then Andes may skip some of the dialog and just give hints on the next step in the principle's procedure. In other words, the procedural scaffolding is there if the student wants it, but they have to ask for it.

When the student asks "what's wrong?" with an incorrect step, then the new Andes will not always given them help on that step. If the student has skipped some critical steps, then Andes will identify which one should be done first and hint it by saying, "Your errors are probably caused by skipping <step>. If you do it, you'll probably figure out what your errors are. If not, ask me for help again." If the student does ask for help again, they will get it in a context which allows the hints to succeed. For instance, they won't actually get hints on fixing a sign error until they have first drawn the relevant vector.

This approach to fading is micro-adaptive in that scaffolding is done only on the part of the strategy whose absence is causing errors or confusion. As students become more competent, they ask for less help and thus receive less scaffolding. Thus, Andes fades the scaffolding without even maintaining a student model.

Andes' micro-adaptive fading should cause more transfer than macro-adaptive fading. If students are forced to follow a specific procedure, then they will only learn how to do the procedure. They won't learn about the errors and confusions that happen if they don't follow it. With Andes' micro-adaptive fading, students learn the value of the procedure the hard way, by making errors and getting lost. This increases the chance that they will use the target strategy when working on their own, thus increasing transfer.

3 Deep Learning

MTTs have sometimes been criticized for failing to encourage deep learning. The following are probably only a few of the criticisms that fall under this category, but they are certainly enough to set a challenging research agenda:

1. If students don't reflect on the tutor's hints, but merely keep guessing until they find an action that gets positive feedback, they can learn to do the right thing for the wrong reasons, and the tutor will never know such shallow learning occurred (Aleven et al., 1999; Aleven et al., 1998).
2. Since the tutor does not ask students to explain their actions, students may not learn the domain's language. Our verbal protocols are replete with domain language misuse. Educators have become increasingly concerned that students learn to "talk science," as that appears to be part of a deep understanding of the science, as well as facilitating scientific writing, working collaboratively in groups, and beginning to participate in the culture of science.
3. The high bandwidth GUI, which asks students to display many of the details of their reasoning, doesn't promote stepping back to see the "basic approach" one has used to solve a problem. Even students who have gotten high grades in a physics course can seldom describe their basic approaches, nor tell when two problems have similar basic approaches (Chi, Feltovich, & Glaser, 1981).

4. Students of quantitative skills, such as algebra or physics problem solving, are usually not encouraged to see their work from a qualitative, semantic perspective, so they fail to induce versions of the skills that can be used to solve qualitative problems and to check quantitative ones for reasonableness. Even physics students with high grades often score poorly on tests of qualitative physics (e.g., Hestenes, Wells, & Swackhamer, 1992).

Many of these objections can be made to just about any form of instruction. Even expert tutors and teachers have difficulty getting student to learn deeply. Therefore, these criticisms of MTT should only encourage us to improve them, not reject them.

There are two common themes in the list above. First, all four involve integrating problem-solving knowledge with other knowledge, namely: (1) principles or rationales, (2) domain language, (3) abstract, basic approaches and (4) qualitative rules of inference. Second, the kinds of instructional activities that are currently used to tap these other kinds of knowledge make critical use of natural language. Although one can invent graphical or formal notations to teach these kinds of knowledge on a computer, they might be more confusing to the students and instructors than the knowledge that they are trying to convey. Moreover, students and instructors are likely to resist learning a new formalism, even a graphical one, if they will only use temporarily.

3.1 Atlas: a Natural-Language Enhancement for Model-Tracing Tutors

We believe that if MTTs are to become more effective at encouraging deep learning, they must use natural language. Therefore, we have begun building *Atlas*, a module that can be added to Andes or other MTT in order to conduct natural language dialogs that will promote deep learning. Atlas uses natural-language generation technology originally developed for CIRCSIM tutor (Freedman & Evens, 1996), the LC-FLEX parser (Rose & Lavie, in press), and the COCONUT model of collaborative dialog (DiEugenio, Jordan, Thomason, & Moore, in press). See (Freedman, Rose, Ringenberg, & VanLehn, 2000) for a description of the system architecture.

Initially, Atlas will support only a simple form of interaction. Most of the time, the students interact with Andes just as they ordinarily would. However, if Atlas notices an opportunity to promote deep learning, it takes control of the interaction and begins a natural language dialog. Although Atlas can ask students to make Andes actions as part of the dialog (e.g., it might have the student draw a single vector), most of the dialog is conducted in a scrolling text window. When Atlas decides the dialog is complete, it signs off and lets the student return to solving the problem with Andes.

The dialogs are called *knowledge construction* dialogs, because they are designed to encourage students to infer or construct the target knowledge. Like a Socratic tutor (Collins & Stevens, 1982), Atlas tries to avoid telling the student what the student needs to know, but it may do so as a last resort.

3.2 Knowledge Construction Dialogs to Teach Principles

So far, Atlas conducts just one kind of knowledge construction dialog. The dialogs are designed to teach a domain principle. They occur when the student has made an error or gotten stuck, and has asked Andes for help. Atlas takes over when Andes would have given its final hint. Instead of telling the student the target principle, it conducts a dialog designed to teach the principle in a Socratic fashion.

The specific dialogs we are implementing were observed in transcripts of human tutors (VanLehn, Siler, Murray, & Baggett, 1998; VanLehn, Siler, Murray, Yamauchi, & Baggett, in press). For instance, consider the target principle "When an object is moving in a straight line and slowing down, its acceleration is opposite its velocity." VanLehn et al. (in press) observed 15 knowledge construction dialogs for this rule. However, they can be reduced to three basic types. One derives the target principle from the definition of acceleration, another uses analogy, and a third shows that the students' belief (that acceleration is in the same direction as velocity) leads to contradictory and absurd conclusions.

Knowledge construction dialogs are often nested. For example, suppose the tutor starts by asking the student for the definition of acceleration. Most students will say, "velocity divided by time," which is almost right, so the tutor corrects it in a subtle way (Graesser, Person, & Magliano, 1995) by splicing in the missing information "Yes, it's *the change* in velocity divided by time." However, if the student's response indicates greater confusion than that (e.g., "It's the derivative of time."), then the tutor may drop into a knowledge construction dialog on the definition of acceleration. Empirically, human tutors seldom nest knowledge construction dialogs more than two deep. If the student seems hopelessly confused, then the tutor may abandon the top level knowledge construction dialog and start a different one by saying, e.g., "Well, forget about the definition of acceleration. Let's try an analogy. Suppose...."

Implementing even one knowledge construction dialog strategy is a major endeavor. Not only must the dialog strategy itself by developed, but types of student responses must be anticipated, each with an appropriate tutorial response. Moreover, the knowledge mentioned in the dialogs must be represented in such a way that the natural language processing modules can both recognize it in the students' contributions to the dialog and render it as fluent, easily understood text for the tutor's contributions. Our progress has been slow thus far, but should pick up as we develop more and more knowledge construction dialog strategies, because we expect them to share many parts.

3.3 Other Knowledge Construction Dialogs

We are in the process of collecting examples of other types of knowledge construction dialogs from a public archive of tutorial dialogs (see http://www.pitt.edu/~circle/Archive.htm) and designing Atlas dialog strategies to encourage deep learning of several different kinds. The following list indicates our plans and is numbered to correspond to the list of deep learning types mentioned earlier:

1. *Avoiding superficial learning*. Critics say that MTT students often learn how to do the right thing for the wrong reasons. That is, they induce conditions for their operators that have roughly the same extension as the correct conditions. To detect such shallow learning, Andes-Atlas should periodically ask students to describe and justify their actions. For instance, if the student enters $F-W-m*a=0$, Atlas-Andes should ask, "What are you doing here?" Hopefully, the

student will answer, "I'm applying Newton's law along the vertical axis." However, if they say, "I'm solving F−W=m*a," then the tutor should probe deeper to see if there is any knowledge behind the algebra.
2. *Using the domain language*. Critics say that MTTs should teach students how to use the language of the domain. For example, if the students, when asked the question above, fail to give a recognizable answer, they should be coached on how to use physics language more accurately. For instance, the tutor might say, "I didn't understand your explanation. Could you say something like, 'I applied <a principle> to <objects> because I wanted <goal>.'? For example, you might say, 'I applied Newton's Second Law to the car because I wanted to find its acceleration.'"
3. *Inducing and using abstract plans*. Critics say that MTTs should encourage students to see the basic approach behind their problem solving and abstract plans from the details. For instance, after students have finished the classic Atwood's machine problem (two blocks hung from either end of a string that is draped over a massless, frictionless pulley), Atlas-Andes could ask them: "What was your basic approach to this problem?" They will hopefully say something like, "I applied Newton's law twice, once for each block."
4. *Connecting qualitative and quantitative reasoning*. MTTs should teach students how to reason qualitatively and how to connect that qualitative reasoning with their quantitative reasoning. For example, the tutor could interject qualitative questions into the student's work, such as "If the acceleration and the tension force are both upward, then increasing the tension should increase the acceleration, right? Is your equation consistent with that fact?" After the problem is solved, the tutor can ask the student to indicate what will happen under other conditions specified qualitatively. For instance, a common question that textbooks ask after students have solved the Atwood's pulley system is, "What would you expect to happen if the two blocks had the same mass? What do your equations say? What would happen if the left block's mass were zero? What do your equations say?"

Andes-Atlas is intended to close the gap between human tutors and MTTs. It provides enrichments to the usual MTT dialog in the form of knowledge construction dialogs. Although it might be possible to provide these enrichments with a GUI solution, the nature of the enrichments makes that unlikely. They clearly call for a language-based solution and that is what Andes-Atlas will provide.

Acknowledgements

This research was supported by Grant N00014-96-1-0260 from the Cognitive Sciences Division of the Office of Naval Research and by Grant 9720359 from the LIS program of the National Science Foundation. We gratefully acknowledge the contributions of the "Andes Alumni," Abigail Gertner, Cristina Conati, Patricia Albacete, Stephanie Siler, Ellen Dugan, Zhendong Niu and David Correll.

References

1. Aleven, V., Koedinger, K. R., & Cross, K. (1999). Tutoring answer-explanation fosters learning with understanding, *Artificial Intelligence in Education* (pp. 199-206). Amsterdam: IOS Press.
2. Aleven, V., Koedinger, K. R., Sinclair, H. C., & Snyder, J. (1998). Combating shallow learning in a tutor for geometry problem solving. In B. P. Goettle, H. M. Halff, C. L. Redfield, & V. J. Shute (Eds.), *Intelligent Tutoring Systems, Proceedings of the Fourth International Conference* (pp. 364-373). Berlin: Spring-Verlag.
3. Anderson, J. R., Boyle, C. F., & Reiser, B. J. (1985). Intelligent tutoring systems. *Science, 228*, 456-462.
4. Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive Tutors: Lessons Learned. *The Journal of the Learning Sciences, 4*(2), 167-207.
5. Chi, M. T. H., Feltovich, P., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science, 5*, 121-152.
6. Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the craft of reading, writing and mathematics. In L. B. Resnick (Ed.), *Knowing, learning and instruction: Essays in honor of Robert Glaser* (pp. 543-494). Hillsdale, NJ: Lawrence Erlbaum Associates.
7. Collins, A., & Stevens, A. (1982). Goals and methods for inquiry teachers. In R. Glaser (Ed.), *Advances in Instructional Psychology, Vol. 2* . Hillsdale, NJ: Lawrence Erlbaum Associates.
8. DiEugenio, B., Jordan, P. W., Thomason, R. H., & Moore, J. D. (in press). The agreement process: An empirical investigation of human-human computer-mediated dialogues. *International Journal of Human-Computer Studies.*
9. Freedman, R., & Evens, M. W. (1996). Generating and revising hierarchical multi-turn text plans in an ITS. In C. Frasson, G. Gauthier, & A. Lesgold (Eds.), *Intelligent Tutoring Systems: Proceedings of the 1996 Conference* (pp. 632-640). Berlin: Springer.
10. Freedman, R., Rose, C. P., Ringenberg, M. A., & VanLehn, K. (2000). ITS Tools for natural language dialogue: A domain-independent parser and planner. In C. Frasson (Ed.), *Proceedings of ITS 2000.* . Berlin: Springer-Verlag.
11. Gertner, A., Conati, C., & VanLehn, K. (1998). Procedural help in Andes: Generating hints using a Bayesian network student model., *Proceedings of the 15th national Conference on Artificial Intelligence* .
12. Gertner, A. S. (1998). Providing feedback to equation entries in an intelligent tutoring system for Physics. In B. P. Goettl, H. M. Halff, C. L. Redfield, & V. J. Shute (Eds.), *Intelligent Tutoring Systems: 4th International Conference* (pp. 254-263). New York: Springer.
13. Gertner, A. S., & VanLehn, K. (2000). Andes: A coached problem solving environment for physics. In C. Frasson (Ed.), *Proceedings of ITS 2000* . New York: Springer.
14. Graesser, A. C., Person, N., & Magliano, J. (1995). Collaborative dialog patterns in naturalistic one-on-one tutoring. *Applied Cognitive Psychology, 9*, 359-387.
15. Hestenes, D., Wells, M., & Swackhamer, G. (1992). Force concept inventory. *The Physics Teacher, 30*, 141-158.
16. Koedinger, K., & Anderson, J. R. (1993). Reifying implicit planning in geometry: Guidelines for model-based intelligent tutoring system design. In S. P. L. a. S. J. Derry (Ed.), *Computers as cognitive tools* . Hillsdale, NJ: Lawrence Erlbaum Associates.
17. Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1995). Intelligent tutoring goes to school in the big city. In J. Greer (Ed.), *Proceedings of the 7th World Conference on Artificial Intelligence and Education* (pp. 421-428). Charlottesville, NC: AACE.
18. Lewis, C. (1981). Skill in algebra. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 85-110). Hillsdale, NJ: Lawrence Erlbaum Associates.

19. Mark, M. A., & Greer, J. E. (1995). The VCR tutor: Effective instruction for device operation. *The Journal of the Learning Sciences, 4*(2), 209-246.
20. McArthur, D., Stasz, C., & Zmuidzinas, M. (1990). Tutoring techniques in algebra. *Cognition and Instruction, 7*(3), 197-244.
21. McKendree, J., Radlinski, B., & Atwood, M. E. (1992). The Grace Tutor: A qualified success. In C. Frasson, G. Gautheir, & G. I. McCalla (Eds.), *Intelligent Tutoring Systems: Second International Conference* (pp. 677-684). Berlin: Springer-Verlag.
22. Merrill, D. C., & Reiser, B. J. (1994). Scaffolding effective problem solving strategies in interactive learning environments. In A. R. a. K. Eiselt (Ed.), *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society* (pp. 629-634). Hillsdale, NJ: Lawrence Erlbaum Associates.
23. Reif, F., & Scott, L. A. (1999). Teaching scientific thinking skills: Students and computers coaching each other. *American Journal of Physics, 67*(9), 819-831.
24. Reiser, B. J., Beckelaar, R., Tyle, A., & Merrill, D. C. (1991). Gil: Scaffolding learning to program with reasoning-congruent representations. In L. Birnbaum (Ed.), *The International Conference of the Learning Sciences: Proceedings of the 1991 Conference* (pp. 382-388). Charlottesville, NC: Association for the Advancement of Computers in Education.
25. Reiser, B. J., Copen, W. A., Ranney, M., Hamid, A., & Kimberg, D. Y. (in press). Cognitive and motivational consequences of tutoring and discovery learning. *Cognition and Instruction.*
26. Reiser, B. J., Kimberg, D. Y., Lovett, M. C., & Ranney, M. (1992). Knowledge representation and explanation in GIL, an intelligent tutor for programming. In J. H. a. C. Larkin, R.W. (Ed.), *Computer Assisted Instruction and Intelligent Tutoring Systems: Shared Goals and Complementary Approaches* (pp. 111-150). Hillsdale, NJ: Lawrence Erlbaum Associates.
27. Rose, C. P., & Lavie, A. (in press). Balancing robustness and efficiency in unification-augmented context-free parsers for large practical applications. In J. C. Junqua & G. V. Noord (Eds.), *Robustness in Language and Speech Technology* : Kluwer Academic press.
28. Shelby, R. N., Schulze, K. G., Treacy, D. J., Wintersgill, M. C., Gertner, A. G., & Vanlehn, K. (in prep.). The Andes Intelligent Tutor: an Evaluation .
29. Shute, V. J. (1993). A macroadaptive approach to tutoring. *Journal of Artificial Intelligence in Education, 4*(1), 61-93.
30. Singley, M. K. (1990). The reification of goal structures in a calculus tutor: Effects on problem solving performance. *Interactive Learning Environments, 1*, 102-123.
31. VanLehn, K. (1988). Student modeling. In M. Polson & J. Richardson (Eds.), *Foundations of Intelligent Tutoring Systems* (pp. 55-78). Hillsdale, NJ: Lawrence Erlbaum Associates.
32. VanLehn, K. (1996). Conceptual and meta learning during coached problem solving. In C. Frasson, G. Gauthier, & A. Lesgold (Eds.), *ITS96: Proceeding of the Third International conference on Intelligent Tutoring Systems.* . New York: Springer-Verlag.
33. VanLehn, K., Siler, S., Murray, C., & Baggett, W. B. (1998). What makes a tutorial event effective? In M. A. Gernsbacher & S. J. Derry (Eds.), *Proceedings of the Twentieth Annual Conference of the Cognitive Science Society* (pp. 1084-1089). Hillsdale, NJ: Erlbaum.
34. VanLehn, K., Siler, S., Murray, C., Yamauchi, T., & Baggett, W. B. (in press). Human tutoring: Why do only some events cause learning? *Cognition and Instruction.*