

Limitations of Student Control: Do Students Know when They Need Help?

Vincent Aleven and Kenneth R. Koedinger

HCI Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

aleven@cs.cmu.edu, koedinger@cs.cmu.edu

Abstract Intelligent tutoring systems often emphasize learner control: They let the students decide when and how to use the system's intelligent and unintelligent help facilities. This means that students must judge when help is needed and which form of help is appropriate. Data about students' use of the help facilities of the PACT Geometry Tutor, a cognitive tutor for high school geometry, suggest that students do not always have these metacognitive skills. Students rarely used the tutor's on-line Glossary of geometry knowledge. They tended to wait long before asking for hints, and tended to focus only on the most specific hints, ignoring the higher hint levels. This suggests that intelligent tutoring systems should support students in learning these skills, just as they support students in learning domain-specific skills and knowledge. Within the framework of cognitive tutors, this requires creating a cognitive model of the metacognitive help-seeking strategies, in the form of production rules. The tutor then can use the model to monitor students' metacognitive strategies and provide feedback.

1 Introduction

Intelligent tutoring systems would not be *that* if they did not provide intelligent help and feedback to learners. The intelligent hint messages and feedback help students reduce unproductive time and thereby to learn more efficiently [Anderson, *et al.*, 1989; McKendree, 1990]. Also, the tutor's explanations, if read carefully, may help students to bridge gaps in their knowledge. Some systems also provide unintelligent, or low cost help, in the form of on-line dictionaries [Shute and Gluck, 1996], or a Glossary [Aleven, *et al.*, 1999].

The common wisdom in the field of intelligent tutoring systems holds that by and large, the system should let students control and organize their own learning

Acknowledgments: This research is sponsored by an NSF grant to the Center for Interdisciplinary Research on Constructive Learning (CIRCLE), a research center located at the University of Pittsburgh and Carnegie Mellon University.

processes; the system should intervene as little as possible [Burton and Brown, 1982] and help should be given on request only. The underlying assumption is that the student herself is a better judge of when help is needed than the system, which does not have much bandwidth to access a students' thoughts, and may not have a complete enough domain model to account for all observed strategies. Thus, in many systems, the help messages are given primarily when the student requests help, for example in Belvedere [Paolucci, *et al.*, 1996] and Sherlock [Katz, *et al.*, 1998].

Recognizing the need for help is a (metacognitive) skill in its own right. It requires that students monitor their own progress and understanding. For example, it requires that students judge whether an error is just a slip and easily repairable, and when an error is due to a lack of knowledge or the result of guessing. Such metacognitive skills are very important, for example, they mediate learning from examples [VanLehn, *et al.*, 1992]. There is evidence that such skills are not mastered by all. There are individual differences with respect to students' metacognitive skills, for example, the ability to explain examples [Chi *et al.*, 1989] or the ability to make productive use of optional on-line tools in a computer tutor [Shute and Gluck, 1996]. Therefore, it is not clear that placing control in the hands of the learner is always the best strategy. There is evidence that higher-ability students do better in (non-intelligent) computer-based environments that offer a greater degree of learner control, whereas lower-ability students do better in more structured environments [Recker and Pirolli, 1992]. Also, students with higher prior ability are better able to judge their need for help after errors than students with lower prior ability, in an intelligent tutoring system with on-demand help [Wood and Wood, *in press*]. These studies suggest that higher-ability students have better metacognitive skills. This means that emphasizing learner control in intelligent tutoring systems may lead to the unfortunate situation that those who need help the most, are the least likely to receive it in time.

Thus, there is ample reason further to study whether the users of intelligent tutoring systems have the metacognitive skills necessary to take advantage of the help facilities that these systems offer. We studied students' use of two types of help facilities of the PACT Geometry Tutor. In this paper, we discuss our findings and the implications for the design of intelligent tutoring systems.

2 Intelligent and Unintelligent Help in the PACT Geometry Tutor

The PACT Geometry tutor is an integrated part of a complete high school course for geometry, developed with guidance from a mathematics teacher. This curriculum emphasizes geometry problem solving (as opposed to proof), including the use of geometry for real-world problems, following guidelines from the National Council of Teachers of Mathematics [NCTM, 1989].

The PACT Geometry Tutor is a cognitive tutor [Anderson, *et al.*, 1995]. It supports guided learning by doing: It monitors students as they solve geometry problems, and

provides hints and feedback. It employs a cognitive model of the skills of an ideal student, represented as production rules. The tutor uses the model, in a process called *model tracing*, to assess students' solutions and to generate hints. The cognitive model is also the basis for student modeling. The tutor maintains estimates of the probability that the student masters each skill in the model, using a Bayesian algorithm called *knowledge tracing*. The information in the student model is used to select appropriate problems and to advance the student to the next section of the curriculum at appropriate times.

In most problems in the PACT Geometry Tutor, students are given a description and a diagram, and are asked to calculate unknown quantities, such as angle measures or segment measures (see Figure 1, window top left). In addition, students must provide an explanation of each solution step, by indicating which geometry theorem or definition justifies it. They can select “reasons” from the tutor's Glossary window, which lists important geometry theorems and definitions, shown in the middle in Figure 1. When students enter a numeric answer or explanation, the tutor tells them whether it is correct or not.

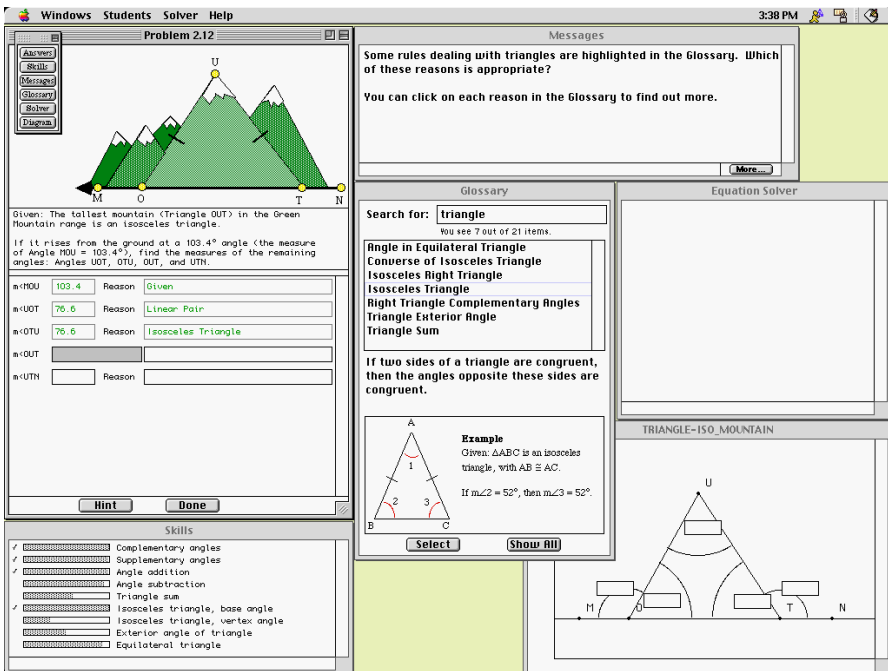


Fig. 1: The PACT Geometry Tutor

The tutor provides intelligent help in the form of on-demand hints and unintelligent, or low-cost, help in the form of a Glossary of geometry knowledge. For each relevant geometry rule, the Glossary contains a description and a short example, as can be seen in Figure 1. Students can search and peruse the Glossary at will. The Glossary was introduced in the PACT Geometry tutor in an attempt to get students to

pay more attention to the rules of geometry (as the reasons behind their actions) and thereby improve their understanding [Alevan, *et al.*, 1999]. Also, the Glossary is much like low-cost sources of information that students are likely to encounter in the real world, such as their own bookshelf, the library, the world-wide web, etc. Teaching students to take advantage of such resources is an important goal in its own right.

The tutor provides intelligent help in the form of on-demand hints. The hints usually have multiple levels, each with increasingly more specific advice. The hints are designed to encourage a general metacognitive strategy: When you do not know something, use an available resource, such as the Glossary, to look it up. Look at what kind of problem you are dealing with, and then look at Glossary rules that are relevant to that kind of problem. For example, if the problem involves a triangle, such as the problem shown in Figure 1, look for rules dealing with triangles (see Table 1, level 1). The next hint highlights a small number of relevant geometry rules the Glossary (Table 1, level 2). Further hints explain which Glossary rule could be used and how it applies to the problem. The last hint of each sequence (the “bottom out hint”) makes it clear what the unknown quantity is, usually by stating an expression, or an equation, (Table 1, level 7).

When students request a hint, this counts as an error, with respect to the tutor's knowledge tracing. That is, the tutor's estimate of the student's mastery of the relevant skill

1. In this problem, you have Triangle OUT. What do you know about triangles that enables you to find the measure of Angle OUT?

2. Some rules dealing with triangles are highlighted in the Glossary. Which of these reasons is appropriate?

You can click on each reason in the Glossary to find out more.

3. The sum of the measures of the three interior angles of a triangle is 180 degrees.

Angle OUT is an interior angle in a triangle. You know the measures of the other two interior angles: Angles UOT and OTU.

4. Can you write an equation that helps you find the measure of Angle OUT?

5. The sum of the measures of Angles OUT, UOT, and OTU equals 180° . So you have:

$$m\angle OUT + m\angle UOT + m\angle OTU = 180$$

6. In the previous hint, you saw that:

$$m\angle OUT + m\angle UOT + m\angle OTU = 180$$

You can replace $m\angle UOT$ by 79 and $m\angle OTU$ by 79. Also, you can use a variable (say, x) instead of $m\angle OUT$. This gives:

$$x + 79 + 79 = 180$$

7. Find the measure of Angle OUT by solving for x :

$$x + 79 + 79 = 180$$

You can use the Equation Solver.

Table 1: A hint sequence generated by the PACT Geometry Tutor.

is debited. This was done for two reasons: The theory behind knowledge tracing requires that a hint request is treated as evidence that the student has not mastered the

given skill. Also, debiting the skill estimate makes it less likely that students complete problems simply by asking for a hint on each step.

Given these tutor facilities, a rational help-seeking strategy may be to try to avoid the penalty that results from errors or hints, and use the Glossary as a first line of defense:

For a step that has not been worked on previously:

If one can find the answer or reason with reasonable certainty (relying on one's memory, not the tutor's help), then enter it.

If one is not sure, use the Glossary: extract a search cue, type the search term into the Glossary, evaluate the rules that are listed.

If all else fails, ask for intelligent help.

After an error:

If one understands what went wrong, then correct the error, without using help.

If one is not sure, use intelligent help.

3 Evaluation Study

In the Spring of 1998, we conducted a study to evaluate the PACT Geometry tutor. The study took place in a suburban school in the Pittsburgh area, in the context of a geometry course based on the PACT Geometry curriculum. Thus, the data pertain to normal use of the tutor in a school. Since the goal of the study was to evaluate the effect of reason-giving, two versions of the PACT Geometry tutor were used: a reason version, which is

	Rate of Glossary use		Rate of Glossary use prior to first attempt		Success rate
	All use	Deliberate use only	All use	Deliberate use only	
Answer Steps	2.7%	2.0%	1.4%	0.8%	54%
Reason Steps	43%	15%	36%	12%	55%

Table 2: Rate of Glossary use, as compared to the success rate. The rate of Glossary use is the percentage of steps for which the Glossary was used. The success rate is the percentage of steps where students got the answer right, without errors or hints.

the version described above, and an answer only version, in which students were not required to provide reasons for their answers. In this paper, however, we present the data for both groups of students together.

The study involved 53 students, enrolled in the course, 41 of whom completed the study. All students participated in four activities: they had classroom instruction, they solved problems on the tutor (they spent 500 minutes, on the average, working on the tutor unit that deals with the geometric properties of angles), and they took a pre-test, and a post-test. The pre-test took place before students started to work on the tutor, the post-test afterwards. The tests involved problems of the same type as those in the

tutor curriculum, and included transfer problems as well. The results indicate that there was a significant improvement in students' test score, attributable to their work on the tutor, in combination with classroom instruction [Alevan, *et al.*, 1999].

4 Use of Unintelligent Help

We analyzed the protocols of students' sessions with the tutor, in order to learn more about their strategies for help use. The protocols, collected automatically by the tutor, contain detailed information about the students' and tutor's actions. We were interested in finding out whether the students followed the strategy for help use, outlined above. We found the following:

- Students used the Glossary on 43% of the explanation steps and used it on 2.7% of the numeric steps (see Table 2). By “step” we mean a subgoal in the problem, or (equivalently) an entry in the tutor's answer sheet, shown on the top-left in Figure 1.
- On numeric answer steps, students used the Glossary about as often in response to errors as they used it before a first attempt at entering a solution (1.3% v. 1.4% of all steps, see Table 2).
- There is little evidence that the students used the Glossary for the more difficult skills, at least not prior to their first attempt at entering a numeric answer. For numeric answer steps, the correlation between Glossary use (prior to first attempt) and skill difficulty was 0.29. Skill difficulty was measured as the success rate for the skill, a measure of performance with the tutor.
- On 0.33% of steps (47 out of 14094), students were apparently able to take advantage of the Glossary to find a correct numeric answer, without errors or hints. In these steps, students went to the Glossary before a first attempt at answering, looked at a Glossary rule that could be applied to solve the step, and perhaps looked at other Glossary items, and then entered a correct answer.

Thus, for numeric steps, the evidence is that students did not follow the strategy for help seeking outlined above. If students followed this strategy, they would consult the Glossary when they anticipated that a step was beyond their capability. They would therefore not make many errors without first consulting the Glossary. But students did not use the Glossary much at all on numeric steps (2.7%) and correctly completed only 0.33% of answer steps with apparent help from the Glossary. They made many errors without first consulting the Glossary: The rate of Glossary use is far below the error rate (see Table 2). The rate of Glossary use is even lower when we count only the steps where the students used the Glossary in a *deliberate* manner. We defined deliberate use as meaning that the student inspected at least one Glossary item for more than 1 second. It is obviously not possible to read and interpret a complete Glossary item in such a short time. However, one second might be enough to recognize as relevant a description that one has read before.

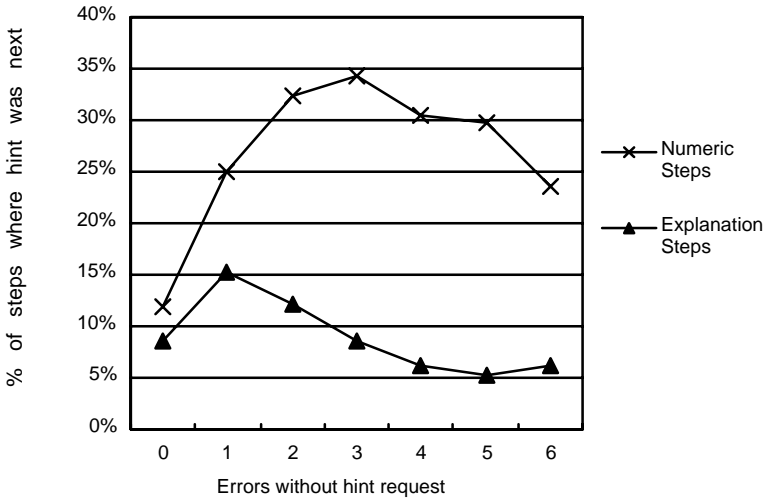


Fig. 2: Frequency of help use after N errors without help—that is, given that a student had made N errors on a step without asking for help, how often was the next action on that step a help request and not another attempt at answering?

Further, if students followed the desired help-seeking strategy, the bulk of Glossary use would occur before a first attempt at entering a step. After that attempt, students could use intelligent help without penalty. Also, students would use the Glossary primarily for more difficult skills. These predictions were not borne out by the data, as shown above. Students used the Glossary primarily to enter explanations (43% of steps). This can in part be explained by the fact that students could enter explanations by selecting from the Glossary. Much of the Glossary use for explanation-giving appears to be rapid selection, but we also saw a considerable amount of deliberate use (see Table 2).

5 Use of Intelligent Help

We also analyzed the student protocols with respect to students' use of the tutor's on-request hints. We found that:

- The students used the intelligent help facility on 29% of the answer steps and 22% of the explanation steps.

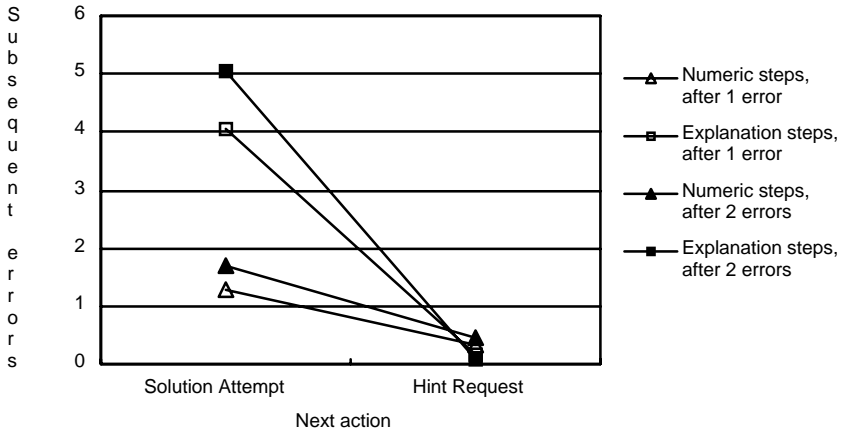


Fig. 3: Subsequent errors on a step, given that the student had already made 1 or 2 errors on that step without asking for a hint, and then attempted another solution (“solution attempt”) or asked for a hint (“hint request”).

- Students used help before their first attempt at answering on 12% of the answer steps and 9% of the explanation steps.
- When the students requested help, they requested to see all hint levels (i.e., they asked for more help until they had seen the bottom out hint) on 82% of answer steps and 89% of explanation steps.
- When students made an error, and if they have not asked for help already, then it is more likely that they would attempt another answer than that they would ask for help, as Figure 2 indicates. This is so regardless of how many errors the student had made already. For example, after making three errors on a numeric step without asking for a hint, students asked for a hint only 34% of the time.

As before, we ask to what extent students followed the help-seeking strategy described above. If students followed the strategy, they would ask for intelligent help in two kinds of situations: when they made an error that they could not fix quickly, and on steps where they had little idea how to proceed. The overall rate of hint use, (29% for numeric steps, 22% for explanation steps), is consistent with this. A close up view of the data reveals however that students often waited too long before requesting a hint, that is, they made too many errors without asking for a hint. According to the desired strategy for help use, students should not make more than one or two errors on a step, before asking for a hint. However, if this was what really happened, the help use graph in Figure 2 would be (close to) 100% after 2 or 3 errors without a hint. Clearly it is not. Further, the fact that students requested to see the bottom out hint in 82% or 89% of all steps with help, indicates that the intermediate-level hint messages were not effective. In short, while there is some evidence that students follow the desired help-seeking strategy, there is significant room for improvement.

We also investigated the immediate effects of the tutor's hints. We found that after one or two errors were made on a step, asking for a hint helped to reduce both the

number of subsequent errors on the same step (see Figure 3) and the amount of time spent to complete the step (see Figure 4). This is evidence that intelligent help aids performance, as was found also in other studies [Wood and Wood, in press; Anderson, *et al.*, 1989; McKendree, 1990].

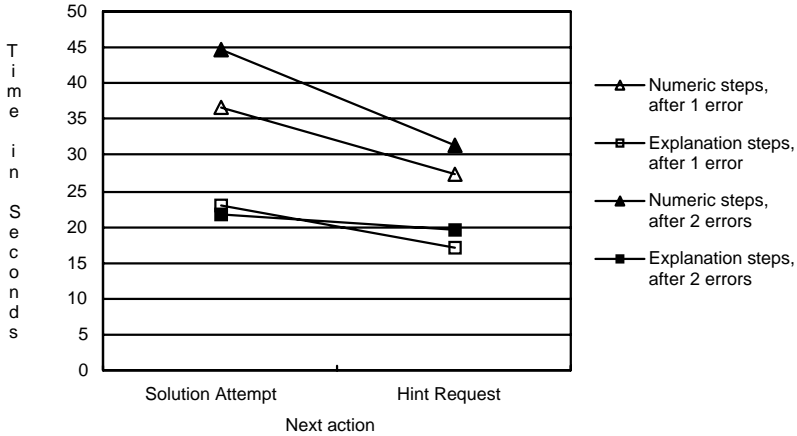


Fig. 4: Time to complete the step, given that the student had already done one of the following four things: made 1 or 2 errors without asking for a hint, and then either asked for a hint (“hint request”) or entered another solution (“solution attempt”).

6 Discussion and Conclusion

Student control is often regarded as a good thing in intelligent tutoring systems. Many systems provide help only on students' request. Thus, the students must decide when and how to use the help facilities, which requires that they make judgments about their own knowledge (self-monitoring) and that they are able to judge when they can benefit from help. But are they able to do so? In order to evaluate the assumption that student control is beneficial, we assessed students' help-seeking strategies in a representative intelligent tutoring system, the PACT Geometry tutor. This tutor provides intelligent help in the form of on-demand hints, and unintelligent help in the form of a Glossary.

The data indicate that students did not use the Glossary very much, contrary to our expectation that they would use it regularly to help with difficult steps. This is surprising, especially if one considers that the students may have been able to avoid errors by using the Glossary and generally seem to be quite aware that the more errors they make, the more problems the tutor will assign. So why would they pass up a source of free help?

This probably says something about the metacognitive skills of the given student population, ninth graders (15-year olds) in a suburban high school. They may not

have sufficient metacognitive skills to judge when they can benefit from Glossary lookup. They may not be very good in judging the difficulty of steps, or at monitoring their own knowledge and understanding. Also, they may not have learned the general strategy to look up things that they do not know. Further, students may lack the mathematical reading ability to interpret the information in the Glossary: statements of definitions and theorems, illustrated with examples. It takes considerable effort to read a rule (even one that has been discussed in class before) and evaluate how it can be applied to the problem at hand.

Students used the tutor's intelligent help facilities far more frequently than they used the Glossary. But they often waited long before asking for a hint, not taking multiple errors on a step as a signal that it is time to ask for a hint. Also, they did not seem to benefit much from the intermediate hint levels. In the vast majority of cases in which they asked for help (82% of numeric steps, 89% of explanation steps), students repeated their help request until they reached the bottom out hint. It may be that in most of these cases, students quite deliberately did not read the intermediate hint levels and read only the bottom out message, which as mentioned pretty much hands them the correct answer. Such hint abuse is undesirable. Learning from being given only the right answer (i.e., the bottom out hint), just as learning from examples, requires that the student constructs an explanation of why the right answer is right. [Anderson, *et al.*, 1989]. The intermediate hint levels may be helpful in this regard. An alternative explanation for the high percentage of bottom out hints is that students find it difficult to read and interpret the tutor's hint messages, just as they seem to find it difficult to interpret the information in the Glossary. In sum, the data indicate that students do not have the necessary metacognitive skills, nor the required mathematical reading ability, to take maximum advantage of on-request help, or the tutor's on-line Glossary.

The tutor should provide more support. It should not always leave it up to the students to judge when they can benefit from using the tutor's on-line help facilities. Rather, the tutor should help students to *learn* to develop effective metacognitive help-seeking strategies (see also [Wood & Wood, in press; Recker and Pirolli, 1992; Conati and VanLehn, 1999]). Further, it should provide support for the interpretation and application of mathematical rules listed in the Glossary.

In order to tutor at the metacognitive level, a cognitive tutor needs a model of the metacognitive strategy to be taught. The model presented in this paper may serve as a starting point, although further study may indicate that it needs to be refined. Within the framework of cognitive tutors, the model could be implemented as a production rule model and be used for model-tracing. It may be useful also to take into account information in the student model to assess whether a student might be over-using or under-using the help facilities, as suggested by Wood and Wood [in press].

For example, with a model of metacognitive strategy in place, the tutor could do much more to help students learn to make effective use of the Glossary. As mentioned, the Glossary is representative of many sources of low-cost help that students will encounter in the real world. Glossary lookup skills learned with the tutor may transfer to the use of the World-Wide Web, for example. Using a model of metacognitive skill, the tutor will be able to help the students in finding relevant

information in the Glossary and evaluating how that information applies to the current problem. When students type a search cue into the Glossary, the tutor could check whether the search will be productive and if not, it could provide feedback and advice. If the tutor is to help students evaluate how a rule in the Glossary applies to the problem at hand, a mapping step needs to be added to the tutor interface, that lets the student point out how a rule is instantiated in the current problem, that is, which geometric objects in the problem (angles, etc.) correspond to objects mentioned in the rule and what conclusions follow. The tutor could provide feedback.

Further, the metacognitive tutor could help students to use the tutor's on-demand intelligent help more appropriately, like the strategy tutor agent proposed in [Ritter, 1997]. For example, it could help the student learn that it is often best to try to use a low-cost source of help before using a source of high cost help. When a student asks for a (high-cost) hint without first using the (low-cost) Glossary, the tutor could interpret this as evidence that the student does not master the desired metacognitive strategy. It could provide feedback saying "Instead of asking for a hint, you might use the Glossary in order to figure this out. It is better to use the low cost help (the Glossary) before you use high-cost help (the tutor's hint messages)." Similarly, if the student did not ask for help when this would seem appropriate, (e.g., after making two or more errors on a step), the tutor could provide feedback, saying: "Usually, when the going gets tough like this, it means that there is a piece of geometry knowledge that you have not mastered yet. It is good to ask for a hint." (As a first step, we have modified the PACT Geometry tutor so that it initiates help after two errors. According to the data presented in this paper, this will reduce the number of errors students make and save them some time.) Similarly, the tutor could protest when a student abuses hints, going straight for the bottom out hint without paying attention to the intermediate levels. When this happens, the tutor could for example require that the student explain the bottom out hint, by constructing a mapping with a Glossary rule. These forms of feedback at the metacognitive level may help students to balance the use of help and errors, which in one study was shown to be an important indicator of success, especially for lower-ability students [Wood and Wood, in press].

To conclude, in order to for intelligent tutoring systems to be truly adaptive, they should help students to develop effective metacognitive skills. The potential payoff is great, since these skills will help students learn in other domains as well. They will help them to become not just better geometers, or mathematicians, but better learners.

References

- Aleven, V., K. R. Koedinger, and K. Cross, 1999. Tutoring Answer Explanation Fosters Learning with Understanding. In *Artificial Intelligence in Education, Proceedings of AIED-99*, edited by S. P. Lajoie and M. Vivet, 199-206. Amsterdam: IOS Press.
- Anderson, J. R., F. G. Conrad, and A. T. Corbett, 1989. Skill Acquisition and the LISP Tutor. *Cognitive Science*, 13, 467-505.

- Anderson, J. R., A. T. Corbett, K. R. Koedinger, and R. Pelletier, 1995. Cognitive tutors: Lessons learned. *The Journal of the Learning Sciences*, 4, 167-207.
- Burton, R. R., and J. S. Brown. An Investigation of Computer Coaching for Informal Learning Activities, 1982. In *Intelligent Tutoring Systems*, edited by D. H. Sleeman and J. S. Brown, New York: Academic Press.
- Chi, M. T. H., M. Bassok, M. W. Lewis, P. Reimann, and R. Glaser, 1989. Self-Explanations: How Students Study and Use Examples in Learning to Solve Problems. *Cognitive Science*, 13, 145-182.
- Conati, C., and K. VanLehn, 1999. Teaching Meta-Cognitive Skills: Implementation and Evaluation of a Tutoring System to Guide Self-Explanation While Learning from Examples. In *Artificial Intelligence in Education, Proceedings of AIED-99*, edited by S. P. Lajoie and M. Vivet, 297-304. Amsterdam: IOS Press.
- Katz, S., A. Lesgold, E. Hughes, D. Peters, G. Eggan, M. Gordin, and L. Greenberg, 1998. Sherlock II: An Intelligent Tutoring System Built Upon the LRDC Tutor Framework. In *Facilitating the Development and Use of Interactive Learning Environments*, edited by C. P. Bloom and R. B. Loftin. Mahwah, NJ: Erlbaum.
- McKendree, J., 1990. Effective Feedback Content for Tutoring Complex Skills. *Human Computer Interaction*, 5, 381-413.
- NCTM, 1989. Curriculum and Evaluation Standards for School Mathematics. National Council of Teachers of Mathematics. Reston, VA: The Council.
- Paolucci, M., Suthers, D., and A. Weiner, 1996. Automated Advice-Giving Strategies for Scientific Inquiry. In *Proceedings of the Third International Conference on Intelligent Tutoring Systems (ITS'96)*, edited by C. Frasson, G. Gauthier, and A. Lesgold, 372-381. Berlin: Springer-Verlag.
- Recker, M. M., and P. Pirolli, 1992. Student Strategies for Learning Programming from a Computational Environment. In *Proceedings of the Second International Conference on Intelligent Tutoring Systems, ITS '92*, edited by C. Frasson, G. Gauthier, and G. I. McCalla, 382-394. Berlin: Springer-Verlag.
- Ritter, S., 1997. Communication, Cooperation and Competition among Multiple Tutor Agents. In *Artificial Intelligence in Education, Proceedings of AI-ED 97 World Conference*, edited by B. du Boulay and R. Mizoguchi, 31-38. Amsterdam: IOS Press.
- Shute, V. J., and K. A. Cluck, 1996. Individual Differences in Patterns of Spontaneous Online Tool Use. *The Journal of the Learning Sciences*, 5 (4), 329-355.
- VanLehn, K., R. M. Jones, and M. T. Chi, 1992. A Model of the Self-Explanation Effect. *The Journal of the Learning Sciences*, 2 (1), 1-59.
- Wood, H. A., and D. J. Wood, in press. Help Seeking, Learning and Contingent Tutoring. To appear in *Computers and Education* (special edition, 2000).