An architecture to combine meta-cognitive and cognitive tutoring: Pilot testing the Help Tutor

Vincent Aleven, Ido Roll, Bruce McLaren, Eun Jeong Ryu, Kenneth Koedinger
Human-Computer Interaction Institute
Carnegie Mellon University
5000 Forbes Ave, Pittsburgh PA 15213, USA

Abstract Given the important role that meta-cognitive processes play in learning, intelligent tutoring systems should not only provide domain-specific assistance, but should also aim to help students in acquiring meta-cognitive skills. As a step toward this goal, we have constructed a Help Tutor, aimed at improving students' help-seeking skill. The Help Tutor is based on a cognitive model of students' desired help-seeking processes, as they work with a Cognitive Tutor (Aleven et al., 2004). To provide meta-cognitive tutoring in conjunction with cognitive tutoring, we designed an architecture in which the Help Tutor and a Cognitive Tutor function as independent agents, to facilitate re-use of the Help Tutor. Pilot tests with four students showed that students improved their help-seeking behavior significantly while working with the Help Tutor. The improvement could not be attributed to their becoming more familiar with the domain-specific skills being taught by the tutor. Although students reported afterwards that they welcomed feedback on their help-seeking behavior, they seemed less fond of it when actually advised to act differently while working. We discuss our plans for an experiment to evaluate the impact of the Help Tutor on students' help-seeking behavior and learning, including *future* learning, after their work with the Help Tutor.

Introduction

A number of instructional programs with a strong focus on meta-cognition have been shown to be effective, for example programs dealing with self-explanation (Bielaczyc, Pirolli, & Brown, 1995), comprehension monitoring (Palincsar & Brown, 1984), evaluating problem-solving progress (Schoenfeld, 1987), and reflective assessment (White & Frederiksen, 1998). These programs were not focused on the use of instructional software. Based on their success, one might conjecture that intelligent tutoring systems would be more effective if they focused more on the teaching of meta-cognitive skills, in addition to helping students at the domain level. A number of efforts have focused on supporting meta-cognition in intelligent tutoring systems (Aleven & Koedinger, 2002; Bunt, Conati, & Muldner, 2004; Conati & VanLehn, 2000; Gama, 2004; Luckin & Hammerton, 2002; Mitrovic, 2003). In some of these projects, the added value of supporting meta-cognition was evaluated. Aleven and Koedinger showed that having students explain their problem-solving steps led to better learning. Gama showed advantages of having students self-assess their skill level. Still, it is fair to say that ITS researchers are only beginning to evaluate the value of supporting meta-cognition in ITSs.

Our research concerns help seeking. There is evidence that help seeking is an important influence on learning (e.g., Karabenick, 1998), including some limited evidence pertaining to learning with interactive learning environments (Aleven et al., 2003; Wood & Wood, 1999). We focus on the hypothesis that an ITS that provides feedback on students' help-seeking behavior not only helps students to learn better at the domain level but also helps them to become better help seekers and thus better *future* learners. We are not aware of any experiments reported in the literature that evaluated the effect that instruction on help-seeking skill has on students' learning and their ability to become better help-seekers in the future.

In order to test this hypothesis, we have developed a Help Tutor, a plug-in tutor agent (Rich et al., 2002; Ritter, 1997) that evaluates students' help-seeking behavior and provides feedback, in the context of their work with a Cognitive Tutor (Koedinger et al., 1997). In developing such a tutor, there are a number of open issues. First, what exactly constitutes good help-seeking behavior? At one level, it seems quite clear that students should work deliberately, refrain from guessing, use the tutor's help facilities when needed and only then (for example, when a step is unfamiliar or after repeated errors), and read problem instructions and hints carefully. However, it is not always easy to know when help-seeking behavior is ineffective and detrimental to learning. For example, Wood and Wood (1999) describe a student who appeared to be requesting help from the system far too often, yet ended up with high learning gains. Furthermore, tutor development requires a detailed model that defines precisely what it means, for example, to work deliberately or to use help only when needed. The creation of such a model is a research contribution in itself. We use the model that is described in (Aleven et al., 2004). Since then it has been modified so that it captures a wider range of students' help-seeking strategies and provides feedback on only the most egregious deviations from reasonable help-seeking behavior.

Second, how should the Help Tutor and the Cognitive Tutor be coordinated, especially when both tutors might have conflicting "opinions" about the student's action? An action can be correct on the domain level but erroneous according to the Help Tutor and vice versa. There are many coordination options, with potentially significant effect on students' learning, and very few guidelines for selecting from them. In this respect, our work has similarities to the work of Del Soldato and du Boulay (1995) whose system, MORE, coordinated the advice of a domain planner and a motivational planner. The domain planner of MORE would typically suggest that a student tackle harder problems as they succeed on easier ones, while its motivational planner might suggest repeating easier problems to improve a student's confidence and level of success.

Third, what kind of architecture can support combined cognitive and meta-cognitive tutoring? Our goal was to use the Help Tutor as a plug-in tutor agent that could be added to an existing Cognitive Tutor (or other tutoring system) with limited or no customization and, importantly, without requiring any changes to the Cognitive Tutor itself.

Although we have initial answers to these questions, we profess not to know yet if they are the right answers. Eventually, evaluation studies will have to settle that issue. There clearly is risk in our approach. Will students take the Help Tutor's advice seriously, even though it probably will not seem as directly helpful to them as the tutor's help at the domain level, to which they are accustomed? The Help Tutor must establish credibility with the students, for example, not intervene at inopportune moments, like the infamous Paper Clip. It also must not give inappropriate feedback or overly increase cognitive load. In this paper, we present our initial answers to the questions raised above and, as preliminary evidence that we are on the right track, we describe our experience pilot testing the Help Tutor with 4 students.

The Help Tutor

The Help Tutor was developed and piloted in the context of the Geometry Cognitive Tutor, an adjunct to a full-year geometry curriculum being used in approximately 350 high schools across the United States. Like all Cognitive Tutors, this tutor monitors students' step-by-step problem solutions using a cognitive model of student problem solving. It provides feedback and, at the student's request, context-sensitive hints related to the problem that the student is solving. For each problem step, multiple levels of hints are available. The hints explain which problem-solving principle applies, how it applies, and what the resulting answer is. The tutor also provides a second form of help, a searchable on-line Glossary with detailed information about the relevant geometry theorems and definitions, which students can browse freely. The tutor keeps track of the student's knowledge growth over time, using a Bayesian algorithm to estimate students' mastery of the skills targeted in the instruction (Corbett & Anderson, 1995). The Cognitive Tutors uses these estimates to select problems, while the Help Tutor uses them to determine the amount of help a student may need on any given step.

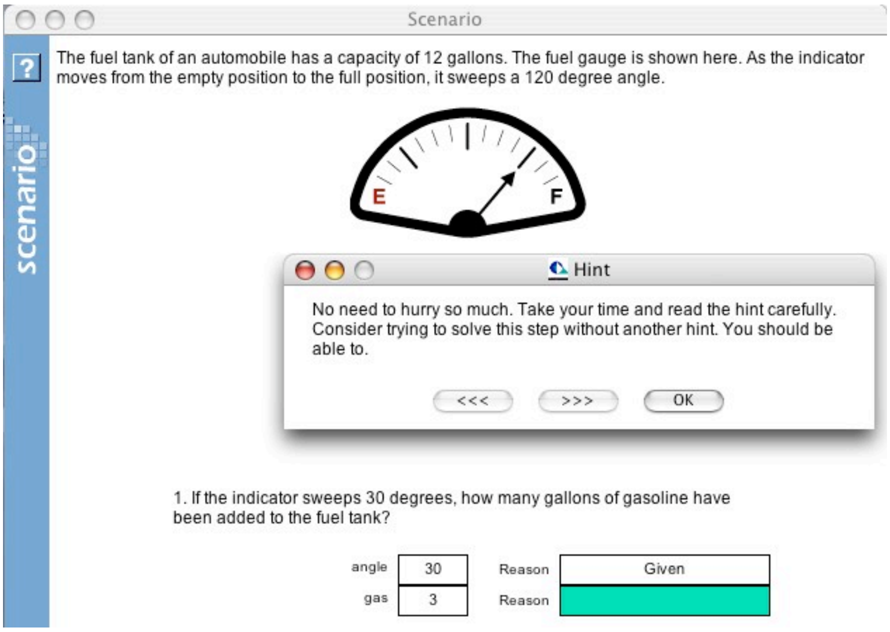

Figure 1: Feedback from the Help Tutor when a student abuses the tutor's context-sensitive hints

The Help Tutor is a Cognitive Tutor in its own right, built using a model of desired help-seeking behavior as a basis. This model, described in more detail in Aleven et al. (2004), is not specific to any given domain, although it is specific to the forms of assistance that Cognitive Tutors offer: feedback, context-sensitive hints, and sometimes a Glossary. According to the model, if a step in a tutor problem is familiar to the student, the student should try it. Otherwise, she should use an appropriate source of help, the Glossary on steps that are at least somewhat familiar, context-sensitive hints on unfamiliar steps. Further, the student should work deliberately: she should spend some minimum amount of time reading problem instructions and deciding what action to take. Similarly, when she requests a hint or uses the Glossary, she should spend at least some minimal amount of time with the hint or Glossary item. When she makes an error and does not know how to correct it, she should take this as a signal that she lacks the relevant knowledge and therefore should use an appropriate source of help. On the other hand, the student should not over-use the help facilities: the more familiar a step, the fewer hints she should use. Looking at too many Glossary items within a given step is also considered to be ineffective help-seeking behavior.

The model is implemented by means of 74 production rules; 36 of these rules capture productive behavior, while the remaining 38 are "bug rules" that capture unproductive behavior. The bug rules enable the Help Tutor to comment on students' unproductive help-seeking behavior, as illustrated in Figure 1. In earlier work (Aleven et al, 2004), we reported that the model identified meta-cognitive errors in 72% of student actions, when applied after the fact to an existing data set. Presenting a message to the student in so many situations is clearly not desirable. Thus, we made the model more lenient by having it focus only on the deviations most negatively correlated with learning. We also improved the model so that it estimates the minimum time it should take the student to read a hint, using research on reading rates (Card, Moran, & Newell, 1983)[1]. In implementing the model, we further had to decide how persistent the Help Tutor should be. That is, to what extent should it force students to follow its advice? For example, when recommending that the student try to solve a given step without a hint, should it withhold its hints until the student convincingly demonstrates that she

[1] A more individual-sensitive improvement we will investigate, as suggested by one of the reviewers, would be to set the minimum hint reading time based on problem solving performance, i.e., students with higher skill levels, as measured by our Bayesian algorithm, and faster problem-solving times may require less hint reading time.

is not capable of solving the step without hints? We decided not to make the Help Tutor insist in situations like this. That is, after the Help Tutor indicates that no hint may be needed, if the student repeats the hint request, the Help Tutor will not protest a second time and the requested hint will be presented. The downside of this approach is that it becomes easier for a student to ignore the Help Tutor's advice.

In integrating meta-cognitive and cognitive tutoring, there must be a way of coordinating the two tutor agents, given that there can be simultaneous, even conflicting feedback from the two sources. For instance, after a hint request by the student, the Cognitive Tutor might want to display a hint, whereas the Help Tutor might want to display a message saying that a hint is unnecessary. In principle, the two types of advice could be kept strictly separate, in space and/or time. That is, the Help Tutor's advice could be presented in a separate window or after the student completed the problem (see e.g., Ritter 1997). However, following the Cognitive Tutor principle "provide immediate feedback on errors" (Anderson et al., 1995), we decided that the Help Tutor feedback would be presented directly after a help-seeking error happens. Further, we decided that the two tutor agents would share a window in which to present messages to the student, rather than give each their own messages window. This was done to avoid the cognitive load that simultaneous messages might cause and to reduce the chance that students would miss or ignore messages from one of the agents. Conflicts between the two tutor agents are handled by a simple resolution strategy (Figure 2). First, after *answer attempts,* feedback from the Cognitive Tutor is given priority over feedback from the Help Tutor. When an answer attempt is correct from the Cognitive Tutor's point of view, it is marked as correct and no error feedback from the Help Tutor is presented, regardless of whether the student followed the desired help-seeking behavior. Coming on the heels of a successful answer, Help Tutor feedback saying, for example, that the student should have taken more time to think or should have asked for a hint instead of trying to answer, is likely to fall on deaf ears, On the other hand, when the Cognitive Tutor deems an answer attempt to be incorrect, it is flagged as incorrect. In addition, an error message may be presented from the Cognitive Tutor or from the Help Tutor. Error messages from the Cognitive Tutor are given priority, since omitting these domain-related messages may reduce the chance that the student can complete the problem. However, such messages are relatively rare in the particular Cognitive Tutor we are using. In practice, the Help Tutor messages are not overridden often.

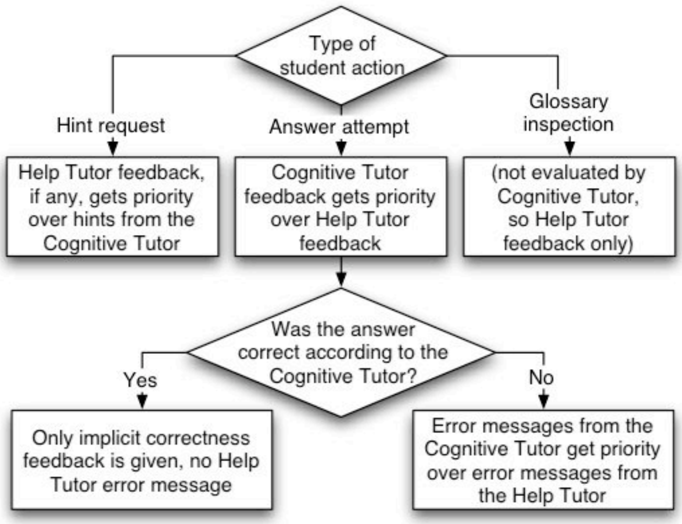

Figure 2: Conflict resolution strategy between the two tutor agents

Second, after *hint requests*, the Help Tutor has priority. That is, if the Help Tutor deems the hint request to be inappropriate, because it is too fast or because the student should be capable of solving the step without (further) hints, the message from the Help Tutor is displayed instead of the requested hint. We hope this will turn out to be an effective way to thwart hint abuse strategies such as clicking through the hint levels at maximum speed until the last hint is reached, a way to induce the tutor to reveal the answer (documented in Aleven et al., in press). However, if the student insists and asks for more hints, the Help tutor does not block them, as discussed previously. Finally, with respect to Glossary use, there are no

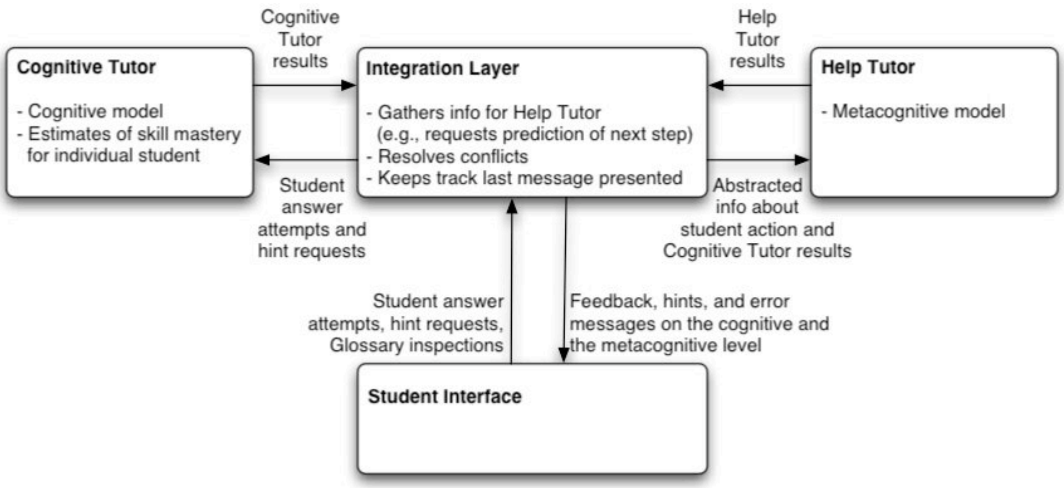

Figure 3: Architecture with two independent tutor agents for combined cognitive and meta-cognitive tutoring

coordination issues, since the Cognitive Tutor does not evaluate students' actions with the Glossary. (Only the Help Tutor does.)

A two-agent architecture

Our goal in developing the Help Tutor was to make it an independent plug-in agent that could be added to existing Cognitive Tutors with little or no customization and without changing the Cognitive Tutor. We realized this objective in a manner similar to the multi-agent approach proposed in Ritter (1997), in which multiple tutor agents are combined in such a way that they maintain their independence. Not only is such modular design good software engineering practice, it is also necessary if the tutor agents are to be easily re-usable. A separate mediator module coordinates the tutor agents. One would typically expect this mediator to be specific to the particular set of tutor agents being combined.

Our architecture, shown in Figure 3, includes two tutor agents: a domain-specific Cognitive Tutor (i.e., an existing tutor, without modifications) and a domain-unspecific Help Tutor. Each of these tutor agents has an identical architecture, the regular Cognitive Tutor architecture, in which a cognitive model is used for model tracing – only their cognitive model is different. An Integration Layer makes sure that the Help Tutor receives all information it needs about the student's interaction with the Cognitive Tutor and resolves conflicts between the two tutor agents in the manner described in the previous section.

In order to evaluate a student's action from the perspective of help seeking, the Help Tutor needs only an abstract characterization of that action, without any domain-specific information, most importantly, the type of the action (attempt at solving a step, hint request, or Glossary lookup), its duration, the student's estimated level of mastery for the skill involved in the step, and, if the action is an attempt at answering, the Cognitive Tutor's evaluation of its correctness. Most of this information is produced in the normal course of business of a Cognitive Tutor. However, some information is needed earlier than it would normally be available, adding to the complexity of the Integration Layer. For example, in order to relate a student's Glossary browsing actions to an appropriate step in the problem, it is sometimes necessary to predict what step the student will work on next, before the student actually attempts that step. To do so, the Cognitive Tutor's model of geometry problem solving is cycled behind the scenes, invisible to the student. The Integration Layer has a number of additional, somewhat mundane, responsibilities, for example, to make sure that the Help Tutor knows which hint or feedback message the student is looking at (i.e., one from the Help Tutor or the Cognitive Tutor), so that it can estimate a minimum reading time. It also makes sure that hint sequences that were interrupted by Help Tutor feedback are resumed at the point of interruption, when the student issues an additional hint request. Such human-computer interaction aspects, we believe, will be an important factor influencing the students' acceptance of the Help Tutor.

A pilot study with the Help Tutor

So far we have evaluated the Help Tutor using existing log files of student-tutor interactions (Aleven et al., 2004). That activity helped in validating the model, but did not produce any information about how students react to its advice. Therefore, we conducted a small-scale pilot study to find out (a) whether students perceive the Help Tutor in a positive light, (b) whether and how the Help Tutor influences their behavior, and (c) whether the Help Tutor intervenes with appropriate frequency. Four high-school students from a public school in a suburban area worked with the Help Tutor. Three of them worked with the tutor for two sessions, one week apart. The fourth student worked with the tutor for the second session only. The students were accustomed to working with the Geometry Cognitive Tutor, as they use it regularly in their classroom, but they were not familiar with the particular curriculum unit involved in the study. The Help Tutor sessions took place during class periods during which the students normally used the Cognitive Tutor, but in a different classroom, separate from the other students in the class, who did not participate in the pilot study. The Help Tutor was modified between the sessions, to fix some problems that were detected during the first session, (mainly usability problems), either by making changes to the model of desired help-seeking behavior or to the Integration Layer.

The results presented here relate to the second session only. Students completed a total of 685 actions (defined as answer attempts, hint requests, or Glossary inspections). The overall ratio of help-seeking errors (according to the Help Tutor) was 16%, ranging from 9% to 24% for the different students (see Table 1). This frequency seems reasonable, since it means that the Help Tutor intervenes once for every six student actions. It suggests that we were successful in making the model a more useful (lenient) standard for help-seeking behavior. (As mentioned above, in an earlier study involving an earlier version of the model, 72% of student actions deviated from the model.) Even more encouraging was the fact that the rate of help-seeking errors dropped from 18% during the first half of the sessions to 14% during the second half. A decrease was observed for all students. These results are only preliminary, as discussed further below. Still, the reduction in error rate is statistically significant (paired-$t=4.0$, $p<0.03$), evidence that the students adapted their behavior to the tutor. Interestingly, the reduction in the error rate cannot be attributed to the students' getting more fluent with the geometry material, since it occurred irrespective of the student's skill level for the given step (high skill: from 16% to 10%; low skill: from 33% to 29%). These numbers are based on the same definition for high/low skill as the Help Tutor uses when evaluating students' help-seeking actions, which in turn are based on the Cognitive Tutor's estimates of skill mastery. Particularly noteworthy is the reduction in errors related to students' help requests, such as asking for hints rapidly and repeatedly. The error-rate for hint requests dropped from 43% during the first half of the students' sessions to 20% during the second half. Previously we found that this behavior is significantly negatively correlated with learning gains and is the most common help-seeking bug (Aleven et al., 2004). Therefore, reducing it was an important goal in building the Help Tutor.

At the end of each session, the students filled out a questionnaire in which they were asked whether they welcomed tutor feedback suggesting that they work slower, ask for a hint, or try without using a hint. They were asked also whether the tutor made these suggestions at appropriate times and with reasonable frequency. One of the four students, though being fond of the Help Tutor after the first session, was quite annoyed by it after the second. She did not like the tutor's suggestions that she reduce the number of hint requests. During the two sessions, this student received more than twice the number of error messages following her hint requests than the other students, due to her faulty use of help. The other three students had

Table 1: Frequency of help-seeking errors during the pilot study

% errors	Student 1	student2	student 3	Student 4	Overall
1st half	20%	27%	10%	15%	18%
2nd half	18%	21%	7%	12%	14%
Overall	19%	24%	9%	13%	16%

a positive opinion about the tutor. All three wanted the tutor to offer suggestions that they work slower and they thought that the tutor presented them at appropriate moments. Two of the three welcomed suggestions from the tutor that they try a step by themselves and thought the tutor presented them with appropriate frequency. The third student thought that these messages are unnecessary.

All in all, these answers are encouraging. They seem to indicate that the Help Tutor's advice was perceived as appropriate and that the Help Tutor did establish some credibility with the students. This is not to say that they always reacted positively at the moment that they received feedback from the Help Tutor. Particularly the "try by yourself" messages were not very popular, as they made it harder for students to get hints. After such a message, one student said: "I hate this tutor!" and another replied: "Because it makes you do the work yourself…" Such comments should probably not be taken as a sign that the tutor was ineffective. It is not unusual for students to complain when working with Cognitive Tutors, even though on the whole, there is clear evidence that the tutors are motivating (Schofield, 1995). Furthermore, if the Help Tutor makes students work harder and does so in an appropriate manner, that may well have a positive influence on students' learning outcomes.

Conclusion

We report on research to investigate whether intelligent tutoring systems can be made more effective if they provide meta-cognitive tutoring, in addition to domain-level tutoring. Our effort is different from other projects in that it focuses on a different meta-cognitive skill, help seeking, and moreover, we focus on *tutoring* a meta-cognitive skill, rather than *scaffolding* it. A key difference is that we do not try to prevent help-seeking errors, but rather, provide feedback when they occur, which we believe will be more effective in getting students to assimilate effective strategies that can and should be used in learning in general.

In developing the Help Tutor, we wanted to make sure that it is a re-usable component that can be plugged in to existing tutors with little or no customization. We achieved this goal by means of an architecture that includes a Cognitive Tutor and Help Tutor as independent agents. This architecture will facilitate the re-use of the Help Tutor in different tutor units and tutors. For example, while we initially implemented the Help Tutor in the Angles unit of the Geometry Cognitive Tutor we are now using it in the Circles unit. This transition was very smooth. In order to use the Help Tutor in conjunction with other units, such as the Similar Triangles unit, some customization will be necessary, due to extra optional tools that students can use in these units, but we do not expect that it will be very burdensome to do so.

The results from a pilot study with the Help Tutor, involving four students, are cause for cautious optimism. The students seemed to adapt to the Help Tutor, as suggested by the fact that over the limited time that they used the Help Tutor, their meta-cognitive error rate went down. Further, in their questionnaires, three of the four students reported that they welcomed the Help Tutor's input and that they found that the Help Tutor gave appropriate feedback. Thus, the Help Tutor seemed to have established some credibility in the eyes of these students. However, these results should be treated with caution. The pilot study was of short duration, involved only a small number of students, and took place outside the real classroom context –in the school itself, during regular Cognitive Tutor lab time, but in a separate room.

We are now conducting a controlled experiment to evaluate the impact of the Help Tutor when it is used in an actual classroom over an extended period of time. This experiment will address key questions that the pilot study left unanswered, such as the Help Tutor's effect on students' learning outcomes and whether it helps them to become better *future* learners.

Acknowledgements

We would like to thank Matthew Welch and Michele O'Farrell for their assistance. This research is sponsored by NSF Award IIS-0308200 and NSF Award SBE-0354420 to the Pittsburgh Sciences of Learning Center. The contents of the paper are solely the responsibility of the authors and do not necessarily represent the official views of the NSF.

References

Aleven, V., & Koedinger, K. R. (2000). Limitations of Student Control: Do Student Know when they need help? In *Proceedings of the 5th International Conference on Intelligent Tutoring Systems, ITS 2000,* 292-303, Berlin: Springer Verlag

Aleven, V., McLaren, B. M., & Koedinger, K. R. (to appear). Towards Computer-Based Tutoring of Help-Seeking Skills. In S. Karabenick & R. Newman (Eds.), *Help Seeking in Academic Settings: Goals, Groups, and Contexts*. Mahwah, NJ:Erlbaum.

Aleven, V., McLaren, B., Roll, I. & Koedinger, K. R. (2004). Toward tutoring help seeking - Applying cognitive modeling to meta-cognitive skills. In *Proceedings of the 7th International Conference on Intelligent Tutoring Systems, ITS 2004,* 227-239, Berlin: Springer Verlag.

Aleven, V., Stahl, E., Schworm, S., Fischer, F., & Wallace, R. M. (2003). Help seeking and help design in interactive learning environments. *Review of Educational Research, 73*(2), 277-320.

Card, S., Moran, T, & Newell, A. (1983). *The Psychology of Human-Computer Interaction.* Mahwah, NJ: Erlbaum.

Conati C. & VanLehn K. (2000). Toward computer-based support of meta-cognitive skills: A computational framework to coach self-explanation. *International Journal of Artificial Intelligence in Education, 11,* 398-415.

Corbett, A.T., Anderson, J.R. (1995) Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User Modeling and User-Adapted Interaction, 4,* 253-278.

Del Soldato, T. & du Boulay, B. (1995). Implementation of motivational tactics in tutoring systems. *International Journal of Artificial Intelligence in Education, 6,* 337-378.

Gama, C. (2004). Meta-cognition in Interactive Learning Environments: The Reflection Assistant Model. In *Proceedings 7th Intern. Conf. on Intelligent Tutoring Systems, ITS 2004* (pp. 668-677). Berlin: Springer.

Gross, A. E., & McMullen, P. A. (1983). Models of the help-seeking process. In J. D. Fisher, N. Nadler & B. M. DePaulo (Eds.), *New directions in helping* (Vol. 2, pp. 45-61). New York: Academic Press.

Karabenick, 1998. *Strategic help seeking: Implications for learning and teaching.* Mahwah, NJ: Erlbaum.

Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. *International Journal of Intelligence in Education, 8,* 30–43.

Luckin, R., & Hammerton, L. (2002). Getting to know me: Helping learners understand their own learning needs through meta-cognitive scaffolding. In *Proceedings of Sixth International Conference on Intelligent Tutoring Systems, ITS 2002* (pp. 759-771). Berlin: Springer.

Renkl, A. (2002). Learning from worked-out examples: Instructional explanation supplement self-explanations. *Learning & Instruction, 12,* 529-556.

Rich, C., Lesh, N.B., Rickel, J. & Garland, A. (2002). A Plug-in Architecture for Generating Collaborative Agent Responses, In *Proceedings of the International Joint Conference on Autonomous Agents and Multi-Agent Systems, AAMAS* (pp. 782-789).

Ritter, S., 1997. Communication, Cooperation and Competition among Multiple Tutor Agents. In B. du Boulay & R. Mizoguchi (Eds.), *Artificial Intelligence in Education, Proceedings of AI-ED 97 World Conference* (pp. 31-38). Amsterdam: IOS Press.

Roll, I., Aleven, V., & Koedinger, K. (2004), Promoting Effective Help-Seeking Behavior through Declarative Instruction. In *Proceedings of the 7th International Conference on Intelligent Tutoring Systems, ITS 2004,* 857-859. Berlin: Springer Verlag.

White, B., & Frederiksen, J. (1998). Inquiry, modeling, and meta-cognition: Making science accessible to all students. *Cognition and Instruction, 16*(1), 3-117

Wood, H., & Wood, D. (1999). Help seeking, learning and contingent tutoring. *Computers and Education, 33,* 153-169.