WHAT DO STUDENTS DO WHEN USING A STEP-BASED TUTORING SYSTEM?

RAJAGOPALAN RANGANATHAN

Arizona State University, Tempe AZ 85034
rrangan5@asu.edu

KURT VANLEHN

School of Computing, Informatics and Decision Science Engineering,
Arizona State University, Tempe AZ 85034
Kurt.Vanlehn@asu.edu

BRETT VAN DE SANDE

School of Computing, Informatics and Decision Science Engineering,
Arizona State University, Tempe AZ 85034
bvds@asu.edu

In analyses of human tutoring, the learning gains are often more strongly correlated with the student's behavior during the tutoring session, rather than with the tutor's behavior. However, we know of only one study that analyzes student reactions when they are working with a *computer* tutor. This motivated us to study the behavior of students when they worked with a step-based physics tutoring system, Andes. As students solved problems on Andes, they gave talk-aloud protocols that were recorded along with their computer screen. Our main finding is that when students asked the tutoring system for help after getting minimal negative feedback, they tended to learn new knowledge or be reminded of something they knew already. On the other hand, if they responded to minimal negative feedback by trying again repeatedly and not asking for a hint, then they tended to fail both to learn and to correct the mistake.

Keywords: Intelligent tutoring system; step-based tutoring system; hinting; feedback.

1. Introduction

Despite their name, tutoring systems are rarely used to replace human teaching. They are instead used to supervise students' practice, such as doing a sequence of math exercises or physics problems. Lectures, text, demonstrations, group work, discussions and many other instructional activities are outside the province of most systems. Tutoring systems are almost always used with problems that have either a well-defined set of correct solutions or at least some well-defined criteria or rubrics for assessing the student's solutions. This allows the tutoring system to give feedback on correctness, suggestions and hints. So the main distinction between a tutoring system and other educational software for students to use while solving problems is that the tutoring system knows

something about the correct solution to the problems and can give the student help based on its knowledge.

The tutoring system's knowledge about the solution can simply be the solution itself, entered by a human author. That is, for every problem, a human author must enter every acceptable solution. Alternatively, a human developer can enter an algorithm for generating or recognizing solutions when given a formal representation of the problem to be solved. If the algorithm is sufficiently complex, then such systems are often called *intelligent* tutoring systems. For instance, the Andes tutoring system discussed here uses a complex algorithm for solving physics problems that is represented as a knowledge-based system with over 350 rules. However, it should be pointed out that the term "intelligent tutoring system" does not have an accepted, precise definition, and many systems that do not have complex algorithms for generating or recognizing solutions are also called intelligent tutoring systems.

A useful and more precise distinction among tutoring systems is based on the granularity of the user interface problem (VanLehn, 2006). *Answer-based* tutoring systems have students entering only the answer to the problem. They are frequently used for simple tasks, such as translating a word from English to German or adding two fractions. When the problems become complex enough that students are likely to use scratch paper to derive an answer, then one can use a *step-based* tutoring system, which provides an interface for users to enter intermediate steps. Typically, a step-based tutoring system allows the users to ask for feedback or hints on the steps they have entered so far, even if they have not yet completed their solution to the problem. The same step-based tutoring system technology is used with the students' task necessarily involves a sequence of steps, such as learning how to configure an operating system or control an unmanned aircraft. The steps in these cases are user interface actions such as flipping a switch or entering a short text string.

Because step-based tutoring systems are used mostly with complex tasks, and it is often impossible or at least unreliable for human authors to enter all acceptable solutions to a task, many step-based tutoring systems are also intelligent tutoring systems. That is, step-based tutoring systems often use a complex, knowledge-based algorithm for generating or recognizing solutions.

Developers of step-based tutoring systems would like to know what students do when they are working with the tutor. This might help them develop tutors that are more effective in helping students learn. Although there have been many studies of human tutoring (discussed below), we are aware of only one study of student behavior while interacting with a step-based tutoring system. In order to find out, at least in a preliminary way, what students do when being tutored by an ITS, we had 10 students solve physics problems using the Andes tutoring system, while talking aloud, for two hours. Initial observations and inferences are presented in this paper, along with suggestions for how to mine further insights from this rich corpus.

One-on-one tutoring by humans is thought to be a much more effective method of instruction than intelligent tutoring systems (Bloom, 1984; Corbett, 2001; Evens &

Michael, 2006; Graesser, VanLehn, Rose, Jordan, & Harter, 2001; VanLehn et al., 2007; Woolf, 2009). Thus, there have been many studies of human tutoring that have attempted to find out why they are so effective (Cade, Copeland, Person, & D'Mello, 2008; Chi, Siler, Jeong, Yamauchi, & Hausmann, 2001; Cho, Michael, Rovick, & Evens, 2000; Core, Moore, & Zinn, 2003; Evens & Michael, 2006; Fox, 1991, 1993; Frederiksen, Donin, & Roy, 2000; Graesser, Person, & Magliano, 1995; Hume, Michael, Rovick, & Evens, 1996; Katz, Allbritton, & Connelly, 2003; McArthur, Stasz, & Zmuidzinas, 1990; Merrill, Reiser, Ranney, & Trafton, 1992; Merrill, Reiser, Merrill, & Landes, 1995; Ohlsson et al., 2007; VanLehn, 1999; VanLehn, Siler, Murray, Yamauchi, & Baggett, 2003). In a recent review, VanLehn (2011) extracted 9 hypothesis from the literature for why human tutoring should be more effective than ITS. He argued that only two of them were viable:

- *Feedback*: Human tutors almost always give immediate feedback on the student's contribution, which may make it easy for the student to self-repair their knowledge.
- *Scaffolding*: Human tutors often provide prompts or hints that push the student to go a little further down a correct line of reasoning. Scaffolding does not present new information to the student, as that would be called a tutorial explanation. It instead gets the student to generate a bit more progress with existing knowledge.

Another theoretical framework comes from Chi et al. (2001), who found support for three hypotheses (a tutor-centered one, a student-centered one, and an interactive one) to explain the effectiveness of human tutoring. Her study suggests that the effectiveness of a tutor depends not only on the pedagogical skills of the tutor, but also on the response that they got from the students i.e. how well do the students construct knowledge after interacting with the tutor. She later generalized her findings and defined a conceptual framework for differentiating learning activities (Chi, 2009). The framework divides the students' learning activities into four types: passive, active, constructive and interactive. For example, reading a book can be considered to be a passive activity; reading a book and highlighting important points can be considered to be active; answering questions can be labeled as constructive; and discussing the questions with a peer or a tutor can be called an interactive activity. By interpreting results of many existing studies, Chi (2009) found considerable evidence for the hypothesis that interactive activities were more effective than constructive activities, which in turn were more effective than active activities, and which were more effective than the passive ones.

These theoretical frameworks agree that it is important to know how students behave when they work with a human tutor. Not surprisingly, analyses of student and tutor utterances (Chi, 1997) have been the dominant method used in research on human tutoring. This suggests that it would be equally illuminating to take verbal protocols of students using an ITS and analyze them. Remarkably, this has been done in only one study known to us: Muldner and Conati (2010) analyzed verbal protocols for use of self-explanation and analogy.

On the other hand, many studies have explored "gaming the system" using methods similar to verbal protocol analysis. Gaming the system is defined as a behavior where the student tries to exploit the properties of the tutoring system to succeed, rather than learn the content (Aleven & Koedinger, 2000; Baker et al., 2006; Baker, Corbett, Koedinger, & Wagner, 2004; Baker, de Carvalho, Raspat, Corbett, & Koedinger, 2009; Baker et al., 2008; Hastings, Arnott-Hill, & Allbritton, 2010; Muldner, Burleson, van de Sande, & VanLehn, 2010; Murray & VanLehn, 2005; Shih, Koedinger, & Scheines, 2008). Several of these studies also measured frequency of off-task behavior (e.g. Baker et al., 2004).

The most often studied form of gaming is "hint abuse." When a tutoring system has a sequence of hints arranged from general to specific, students sometimes rapidly click through the general hints until they get to the last hint, which is called the bottom out hint. It states explicitly what the students must do in order to enter the next correct step. Although some students self-explain the bottom out hint (Shih et al., 2008), most simply copy from the bottom out hint.

Because hints can be abused, some researchers have begun to question the value of hints (Muldner, Burleson, van de Sande, & VanLehn, 2011; Ringenberg & VanLehn, 2006). They have wondered if reducing the student's use of hints, perhaps by providing alternative scaffolding such as a worked example, might be better.

In summary, improving an ITS might be easier if we knew how students used the system and in particular how they used its scaffolding. Unfortunately, most existing studies of student behavior while working on an ITS were confined to analyzing the gaming behavior of the students and did not explore in general on how students used the system. This research is our initial attempt to study the overall behavior of students while working on a computer-based tutoring system.

In this initial study, we conducted a descriptive analysis of the data rather than one that focused on testing hypotheses. Because learning with human tutors seems to occur mostly at impasses (VanLehn, 1999), we focused on episodes when students made an error and struggled to fix it. We characterized their behavior in two ways: (1) what they did during the episode and (2) what was the outcome of the episode. Learning something new was found to be just one possible outcome.

2. Methods

2.1. *Participants*

Ten paid volunteers participated in the two-hour study. These volunteers were undergraduate students recruited from a large introductory physics course. Prior to this study they had already taken the midterm exam in their physics class that covered the relevant topic, Work and Energy. Thus, they had some experience with the topics covered in the experiment but did not have full mastery of the subject. None had used Andes before.

2.2. *Materials*

For the study, we used Andes3, which is similar to Andes2 (VanLehn et al., 2005) with an improved user interface. The Andes3 user interface is shown in Figure 1. The user interface of Andes3 gives the users the same feel they would get when they are solving problems in pen and paper. They can draw bodies, axes, define quantities, and write equations the same way they would do on a paper, except that Andes understands what they wrote and gives immediate feedback on each entry by turning it green if it is correct and red if it is incorrect. Andes can also provide hints when asked, and sometimes gives unsolicited hints as well.

Unlike paper, Andes requires students to follow certain conventions on what can be written. For instance, it requires that dimensional numbers have units and that variables be defined before being used in an equation. These conventions clarify the solution and may be required, for instance, if the solution were hand-graded by an instructor. Andes enforces these requirements both because its designers believe that doing so improves student learning and because Andes can do a better job of interpreting the student's solution. As the data we collected made clear, much of students' learning was just learning to follow these conventions. On the one hand, the learning of physics conventions might be fundamentally different from the learning of physics principles, so perhaps our observations will not generalize. On the other hand, it provided us with many more opportunities for studying learning than we would otherwise have had.
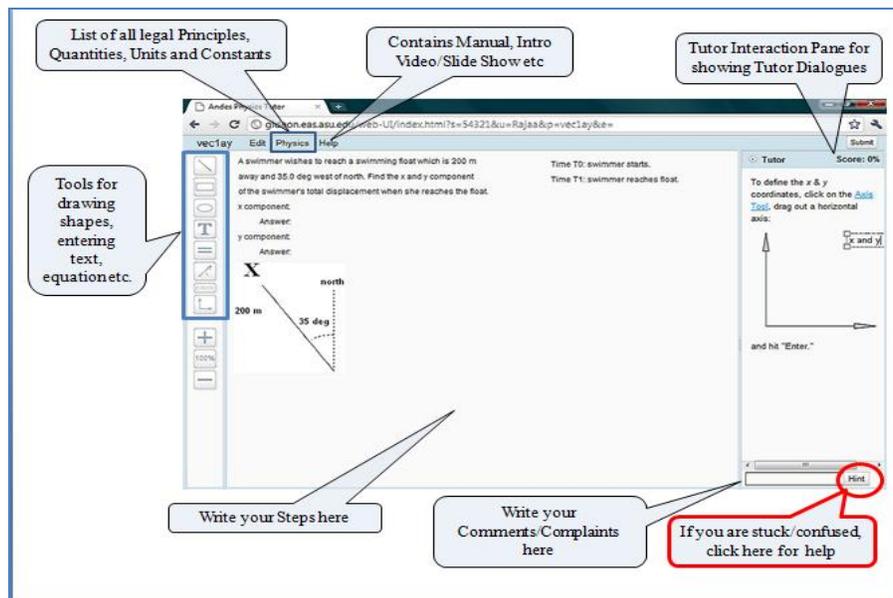
Figure 1. The Andes3 User Interface, with callouts explaining how to use it.

A woman pulls a suitcase with a strap which makes an angle
of 37.0 deg with the horizontal. The suitcase has a mass
of 15.0 kg, and has wheels on it so that it can be considered
to move without friction. If the tension in the strap is 80.0 N,
what work is done on the suitcase by the strap when it is
displaced 2.50 m horizontally?

Answer:

Time T0: pulling starts.
Time T1: pulling ends.

37 deg
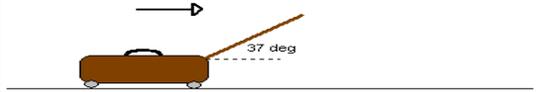
Figure 2. Problem 1.

A man pushes a shopping cart of total mass 55.0 kg up a 5.00
deg ramp. He pushes the cart with a downward force of
130 N at 20.0 deg below the horizontal. If his son pushes
backwards on the cart with a force of 8.00 N down the slope,
what is the net work done on the shopping cart in moving it
22.0 m? Ignore the effects of friction on the shopping cart.

Answer:

Time T0: pushing begins.
Time T1: pushing ends.
g is the acceleration of gravity on earth
g=9.8 m/s^2

20 deg

5 deg
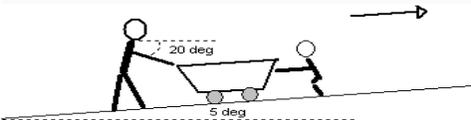
Figure 3. Problem 2.

Students were given a sequence of physics problems to solve. All the problems used the concepts of Work and Energy. Examples of problems are shown in Figure 2 and Figure 3. The Andes Help menu contained brief summaries of the relevant concepts and principles, including the relevant equations. Students did not have a textbook or any other source of physics information other than Andes.

2.3. *Procedure*

After obtaining informed consent, students were trained on the user interface and the Andes' conventions. First they watched a short video that showed a step by step solution for a simple vector problem while explaining the tools and conventions of Andes. Next they were walked through solving the same vector problem. That is, they were required to solve the problem with instructions given at every step. The video and the walk-through comprised the whole user interface training. The video remained available under the "Help" menu item at the top of the screen, but students seldom referred back to it. The walk-through was essentially unavailable, as accessing it would require closing the current problem, opening the walk-through, then reversing the process to return to the original problem.

The students' remaining time was spent solving physics problems. Students were asked to solve as many problems as they could in the remainder of the two hour period. On average, they completed 5.9 problems and had started on one more when time expired. They all worked through the same sequence of problems.

As they solved problems, students were asked to provide information on their cognitive and affective state by talking continuously about what they were thinking and how they were approaching the problem. In addition, the experimenter was nearby and would occasionally intervene at the student's request or if the student stopped making progress for an extended period of time; these instances were recorded. The students' computer screens and their speech were recorded as screen-capture video files. The screen capturing and audio recording were done using Camtasia Studio. The coding was done using Elan.

The data analysis consisted of first segmenting the screen videos into episodes, then coding each episode, and finally counting the codes and aggregating. We worked directly with the screen videos and did not transcribe them. Because these results are intended to be descriptive rather than a test of a hypothesis, we did not use a second coder in order to check for inter-coder reliability.

2.4. *Coding episodes*

In order to segment the video into episodes, we used the fact that Andes provides immediate feedback by coloring a student's entry green if it is correct and red if it is incorrect. An episode was defined as starting from an incorrect (red) entry and ending when that entry was either entered correctly (green) or deleted.

An example of an actual episode is given in Table 1 and Figure 4. The student, who was working on the problem shown in Figure 2, made an incorrect entry (action 1 in the Table). This created a red text "d = 2.50 m" that stayed on the screen until action 6, when the student clicked on the red text, typed nothing, and pressed the Enter key, thus turning the text green and ending the episode. In between, the student defined a variable, thus

Table 1. Student episode.

	Student action	Tutor response
1.	"d = 2.50 m" (Using equation tool)	Red Color (Incorrect Entry) and Unsolicited Hint: "Undefined Variable d."
2.	"Let d be the displacement of the crate" (Using text tool)	Red Color (Incorrect Entry)
3.	Delete "Let d be the displacement of the crate"	None
4.	Click on Hint Button	Hint Given: "Variables must be defined before being used in an equation. Vectors are defined with the vector tool and Scalars are defined with the Text tool"
5.	"Let d be the displacement of the crate" (Using vector tool)	Green Color (Correct Entry)
6.	"d = 2.50 m" (re-entered by clicking on the equation to edit it, making no changes and pressing Enter key)	Green Color (Correct Entry)

Figure 4. A student episode.

fixing the error that made "d = 2.50 m" incorrect. Defining the variable was initially done incorrectly (action 2) and deleted (action 3), but the student asked for a hint (action 4) which enables the student to enter a correct definition (action 5). Although actions 2 through 5 fit the definition for an episode, we decided to keep the coding simple and not use nested episodes, so actions 2 through 5 were not considered an episode, but were instead considered to be part of the episode that starts with action 1 and ends with action 6.

An example of an episode that ends with the student deleting his entry is shown in Table 2. The student has just started working on the problem, *"A man pushes a 20.0 kg crate across a frictionless floor with a horizontal force of 24.0 N. What work is done by the man on the crate in displacing it by 5.00 m?"* Action 1 is incorrect, thus starting an episode. The episode ends at action 7, when the initiating entry is deleted. This episode

Table 2. An episode involving deletion.

	Student action	Tutor response
1.	"W=F*d" (Entered using the equation tool)	Red Color (Incorrect Entry), Unsolicited Hint: Undefined variables W,F and d.
2.	"W: Work" (Entered with the text tool)	Green Color (Correct Entry)
3.	"F: Force" (Editing the above entry)	Red Color (Incorrect Entry)
4.	"F: 24 N" (Editing the above entry)	Red Color (Incorrect Entry)
5.	"F: 24.0 N" (Editing the above entry)	Red Color (Incorrect Entry)
6.	Delete "F: 24.0 N"	None
7.	Delete "W=F*d"	None

indicates again how several red entries can be part of an episode without themselves starting an episode.

2.5. *Coding conventions*

We defined three categories for coding the episodes: (1) Outcome, (2) Behavior and (3) Struggling. An episode could have at most one code from each coding category. Each subsection below describes a coding category and its codes. Although the episodes were segmented using log data, the categorization required listening to the students' verbal protocols. In particular, coding was not done from transcripts but was done directly from the screen recordings, which include an audio track.

2.5.1. *The Outcome coding category*

The Outcome coding category represents the final products of an episode. That is, did the student seem to have learned something and use it to correct the initiating entry, and if not, how did the student deal with the incorrect entry that started the episode? This coding category had 8 codes, which are listed below and explained afterward: Learning new physics; Learning Andes conventions; Applying existing knowledge; Guessed the correction; Guessing or applying knowledge; Copied a bottom-out hint; Deleted the red entry; and Miscellaneous correction.

Learning New Physics: This code represents that the students learned some physics that they appear not to have known earlier. Their spoken comments often indicated when they were learning new physics, which can be difficult to determine when using only log data.

For example, Figure 5 shows a scene where the student must find the magnitude of the final velocity (v), given the initial velocity (u), the acceleration (a) and the displacement (d). Now, this can be done directly using the equation $v^2 = u^2 + 2*a*d$. The student had defined all the necessary quantities for solving this problem but he seems unaware of the equation. This equation is not always taught, because it can be derived from other kinematics equations, which may be why this student was unfamiliar with it. After entering an incorrect entry and fussing around for a while, he opened the Andes
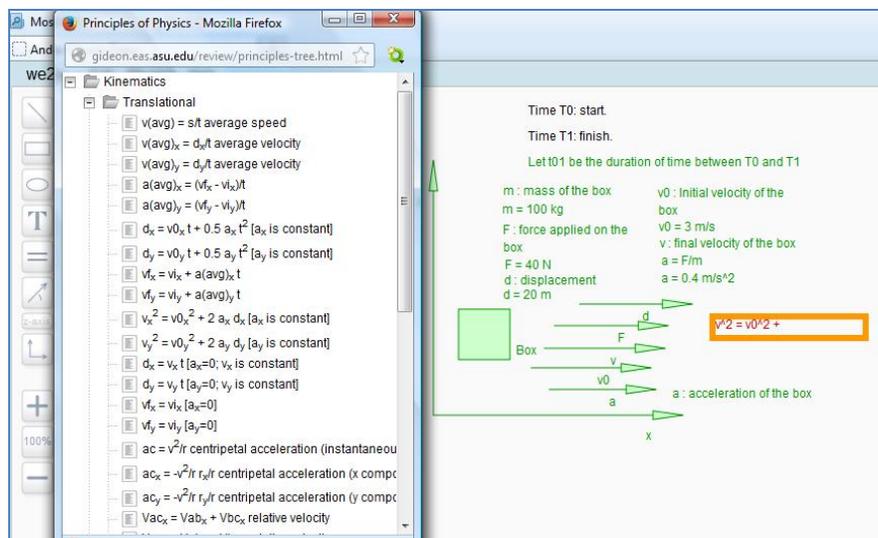
Figure 5. Learning new Physics.

Help menu, navigated to the list of principles (shown on the left side of Figure 5) and saw the equation. He remarked, "Oh, $v^2 = u^2 + 2*a*d$." His evident surprise, which was easily discerned in the audio track of the screen capture video, implies that this was a new equation to him. After the state shown in Figure 6, he applied the equation and the entry turned green, ending the episode. In this episode, he seems to have learned a new kinematics equation, so it received the code, "Learning new physics."

It should be pointed out that we did not expect physics learning to be frequent, because the students had already studied the appropriate physics content to some extent.

Learning Andes conventions: As mentioned earlier, Andes expects students to follow certain conventions while solving problems. For instance, it expects the students to define a variable before using it in equations. Moreover, vectors must be defined using the vector tool; scalar quantities must be defined using the text tool; and equations must be written using the equation tool. This coding category represents episodes in which students appear to learn that Andes requires complying with a convention.

Figure 6 shows an example from a student who is working on the problem shown in Figure 2. The student defined displacement using the text tool. The first hint, "Note that displacement is a vector quantity," doesn't seem to help the student, perhaps because the student was not yet aware of its implications for selecting a tool. The student clicked on the "Explain more" button, which told him to use the vector tool for defining the displacement since it is a vector quantity. The hint has hyperlinks in it to help students who don't know what some of the terms refer to, but this student didn't click on any of them. Next, the student deleted the original red entry and used the vector tool to define

Figure 6. Student uses text tool, gets hint and uses vector tool instead.

displacement. In this process of correcting his red entry, he seems to have learned something new about Andes conventions. Note that this learning occurred despite the fact that this convention was mentioned in the introduction video shown to the students before they start solving.

Applying existing knowledge: This code represents episodes where students already knew something, but forgot to apply it. Students often mention this, even when they must ask for a hint and it reminds them of the knowledge they "knew" but failed to apply.

For example, consider Figure 7. Here the student has solved a problem completely, but while entering his final answer in the answer box, he forgot to put units to the answer, which is required according to the conventions followed in Andes. He got an unsolicited

Figure 7. Applying existing knowledge: Forgetting to put units on a quantity and Andes' reminder.

hint, "Forgot to put units on a quantity." He quickly corrected the answer, which suggests that he had merely forgotten about the Andes convention. The entire episode takes less than 8 seconds.

Guessed the correction: This code represents episodes in which students corrected the red entry that started the episode after a rapid sequence of failed attempts. An example is shown in Figure 8. The student was trying to define a horizontal force, but drew it slightly tilted so it turned red. Instead of taking a hint, he kept drawing vectors and on his 4th attempt, the vector turned green. Note that when drawing a vector, Andes displays the inclination of the vector in degrees. Thus, had the student known that the vector should be 0 degrees exactly, the student could have wiggled the vector as it was being drawn until Andes displayed 0. Apparently, this student didn't know that Andes insists on exact directions.

Figure 8. An example of guessing the direction of a vector.

Guessing or applying knowledge: There are some episodes in which it is not clear even from the audio/video if they were guessing or using the knowledge that they already have. All such episodes fall into this category.

Copied a bottom-out hint: Help is provided in Andes as a sequence of hints, starting from the most general hints and moving toward more specific ones. The last hint in this sequence, the bottom-out hint, tells the next step to the student. When students just copied the bottom-out hint without providing any evidence of self-explanation of the step, then the Outcome category for the episode was "copying a bottom-out hint." This often (but not always) occurred after students had exhibited the classic help abuse behavior of clicking rapidly through the earlier hints into to get to the bottom out hint.

Deleted the red entry: This represents those episodes in which instead of correcting the red entry, the student chooses to delete it. An example of this behavior was given in Table 2.

Miscellaneous corrections: This code for the Outcome coding category is for episodes which do not fit into any of the above codes.

2.5.2. *The Behavior code*

On encountering an incorrect entry the student is expected to correct it. There are different ways in which he can attempt to correct the incorrect entry. This category represents them. This category had 8 codes, which are listed below and explained in the rest of the subsection: Asked for all hints, rapidly; No hints, much guessing; Asked for some hints; Referred to example; No hints, little guessing; Asked expert for help; Andes Defect; Others.

Asked for all hints, rapidly: This is the code for hint abuse, in which students rapidly keep clicking on the hint button till they reach the bottom-out hint. They click so rapidly that they are probably not reading any of the hints except the bottom out hint. The moment they reach the bottom-out hint, the step is right in front of their eyes. If they copy the step without any signs of self-explanation or other thought, then the Outcome category is "copying bottom out hint." Figure 9 shows the initial state of an example of an episode whose Behavior code is "asked for all hints rapidly" and whose Outcome code is "copied bottom out hint."

The goal of this problem is to find the work done (say, W) when given the force (F), the displacement (D) and the angle (theta) between F and D. This is as simple as applying the equation W=F*D*cosine (theta). At the point shown in Figure 9, the student had defined the force and was attempting to define the displacement, but he used the equation

Figure 9. Hint Abuse - Initial problem scene.

tool instead of the vector tool. At this point, he gets an unsolicited hint ("Undefined variables: d") and then clicks rapidly through the following hint sequence:

- "Why don't you continue working on entering the given value of the magnitude of the displacement of the suitcase between T0 and T1."
- "Try introducing a variable for the magnitude of the displacement of the suitcase between T0 and T1."
- "Notice that between T0 and T1, the suitcase is moving along a straight line."
- "Whenever an object moves in a straight line, the displacement vector is parallel to the direction of the motion."
- "Because the suitcase is moving in the direction 0 deg, use the Vector Tool to draw a displacement vector in the direction 0 deg between T0 and T1."

When he reaches the last hint, he uses the vector tool to define displacement. We chose this example to illustrate that sometimes the hints given by Andes are not relevant to the student's problem, which was that the student used the wrong tool to define displacement. This hint sequence was aimed at someone who either didn't know that displacement needed to be drawn or had gotten the direction wrong.

No hints, much guessing: Sometimes when students get a red entry, instead of asking for help, they keep editing the answer until it turns green (correct). This is sometimes referred to as hint refusal. Consider the episode in Figure 10. Here the student was trying to define a displacement and made two mistakes. One was trying to define displacement as distance and the other was using the equation tool instead of a vector tool. The student got an unsolicited hint which he seemed not to understand. Although he could have clicked on "Explain more," he instead kept attempting to correct his mistake. He made
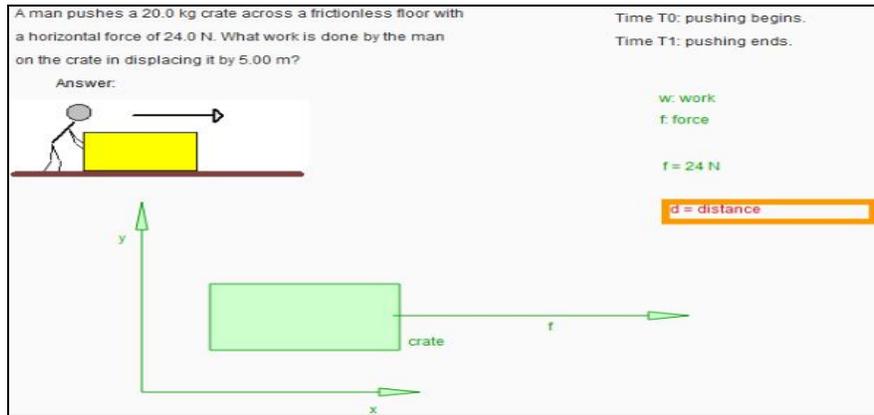
Figure 10. Initial state for an example of hint refusal.
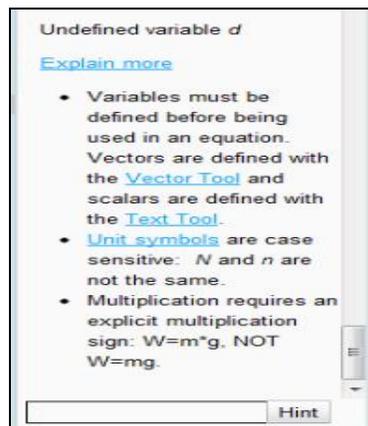
Figure 11. The hint that finally worked.

many attempts using incorrect tools and definitions. Eventually he got frustrated, said so and decided to read some of the hints provided. He finally got a hint (Figure 11) that gave him the idea that he was supposed to use the vector tool and "displacement" instead of "distance." He consequently entered a correct step. Overall, he made more than ten attempts to correct his mistake. All this could have been avoided if he had asked for more hints in the initial stages instead of guessing so many times.

Asked for some hints: This code represents those cases in which the student gets a red entry and rectifies it using the hints given by Andes. Figure 7 showed an example. There, the student forgot to put units for the work done and he got an unsolicited hint saying "Forgot to put units on a number." On seeing this, he realized his mistake and corrected it. Apparently, the hint was effective in allowing him to correct his mistake.

Referred to example: As already mentioned, the very first Andes problem given to all the participants of this study was a walk-through problem. That is, it gave step by step instructions to the student. When students ignored or misunderstood the instruction and made any mistake, all they had to do is to look at the step by step instruction. Figure 12 shows an example. Instead of displacement, the student used the term "distance" which is incorrect. In the pane entitled "Tutor," we can see the step by step instructions for solving this problem. There is a step that tells him to draw a vector and label it as "d is the displacement of the swimmer." The student reads this and corrects his mistake. Thus, this episode received the Behavior code of "referred to example."

No hints, little guessing: This category represents cases where after getting an incorrect entry, the student corrected it almost immediately without receiving any kind of help.

Asked expert for help: There are some episodes in which students were stuck at some point without knowing how to proceed or without knowing how to rectify a red entry. If they got very frustrated and were on the verge of giving up, the experimenter provided

Figure 12. Referring to the example.

some help. In most of these cases, students were stuck because of some software bugs or bad hints.

Andes Defect: These represent episodes where it is impossible for the students to correct their mistake due to some defect present in Andes. This code is applicable only to those episodes that involved the deletion of a red entry, because the only way to proceed was to delete the entry and work around it.

Others: This code is for episodes that do not fall into any of the above categories. For instance, when students simply deleted a red entry without any guessing or asking for hints, then the episode was classified as "Other."

2.5.3. *The Struggling coding category*

In the process of solving problems, whenever the student made a mistake and received a red entry, they sometimes realized their mistake on their own and corrected quickly. Sometimes the unsolicited hints gave them a clue on what could have gone wrong and then they manage to rectify it themselves rather quickly. However, there were times when they struggled for quite some time either making many attempts on their own or asking for help continuously and trying to correct their mistake. Some of these cases were due to unclear hints or bugs in Andes, and at other times it was the students' mistake in that they didn't ask for help when they were stuck at some place for a long time. (This phenomenon is known as Hint Refusal). Therefore, we defined a coding category that represents the effort of the students in correcting their mistakes. For simplicity, we used only two values: yes or no.

2.6. *Example episode codings*

In order to illustrate some more details in the coding scheme, we will include some examples showing the division into episodes and their corresponding codes.

Consider the episode shown in Figure 13. Here, the student has defined everything needed for solving the problem, and just has to write the equation for the net work done on the shopping cart. One equation for work, which this student tried to use, is $W=F*d*cos(Theta)$, where F is the force exerted by the man on the cart, d is the displacement of the cart and *Theta* is the angle between F and d. Many students do not have a clear idea of what *Theta* represents, and end up making mistakes here. In this case *Theta* is in fact 25 degrees, whereas most of the students write it as 20 degrees or 340 degrees, thus leading to a red entry. But, in this case, the student made use of the hints well, and landed up in the Principles menu (Figure 14), which gave an alternate equation for work as $w=fx*dx + fy*dy$. On using this form of the work equation, he got the correct step. Therefore, the code for the Outcome category is "Learning new Physics." The code for the category "Behavior" is "Asked for some hints," since it was the hints that made him see an alternate form of the work and energy equation. He went through the hints very smoothly, therefore the code for the "Struggling" category is "No."

A man pushes a shopping cart of total mass 55.0 kg up a 5.00 deg ramp. He pushes the cart with a downward force of 130 N at 20.0 deg below the horizontal. If his son pushes backwards on the cart with a force of 8.00 N down the slope, what is the net work done on the shopping cart in moving it 22.0 m? Ignore the effects of friction on the shopping cart.

Answer:

Time T0: pushing begins.
Time T1: pushing ends.

F: force of man
Fg: force of gravity from the Earth
f: force of son

d = 22.0 m
F = 130 N
Fg = (55.0 * 9.8) N
f = 8.00 N

d: displacement of cart

w = work done by son

Wtot = total work
W = work done by man

Figure 13. An episode where student learned a new principle.
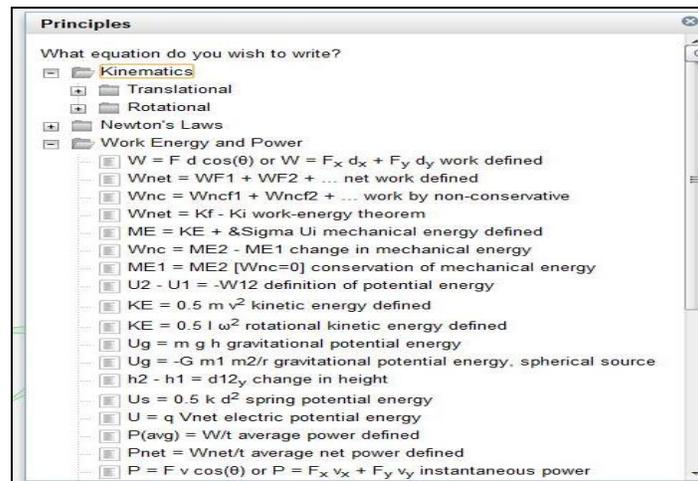
Figure 14. The Andes menu of physics principles.

Next, consider the episode shown in Figure 15. Here, the student was working on the walk-through problem where the system gives him the step by step instructions for solving the problem. The student should have written *d_y = d*sin (125 deg)*, but instead he wrote it *y=d*sin (125)*, which leaves out both "*d_*" and "*deg*". Now, in this case, on the right hand side, an unsolicited hint was given to him, which says that variable *y* is undefined. Also, the correct way to write the equation is shown in the instructions. He looked at the example step and corrected his equation. He did not spend much time, since
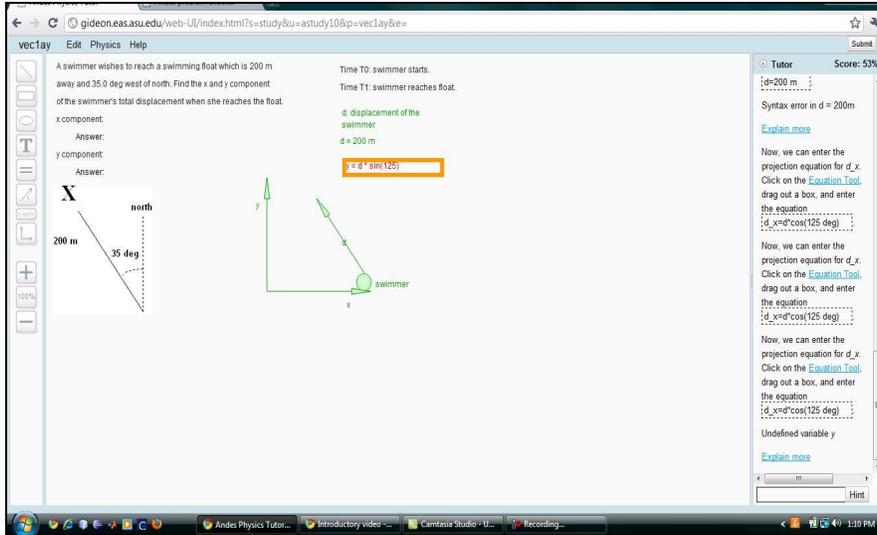
Figure 15. An example of learning an Andes convention.

the answer was right before his eyes. Therefore the code for the Outcome is "Learning Andes Conventions," the code for Behavior is "Refer to Example," and the code for Struggling is "No."

Next consider the episode not shown in a figure. Here, the student wrote the equation as *W=F*d*cos(330)* instead of the correct equation *W=F*d*cos(330 deg)*. He forgot to mention "*deg*" which is the unit for degrees. Because he already had written many equations prior to this, he realized his mistake himself and quickly corrected the error. Therefore the Outcome code is "Apply Existing Knowledge," the Behavior code is "No hints, little guessing," and the Struggling code is "No."

3. Results

A total of 283 episodes were coded for the 10 students, or 28.3 episodes per student on average. Table 3 displays all the episodes and how they were coded along the most important two categories. The columns show the behavior code and the rows show the outcome code.

In order to draw some conclusions based on proportions, we aggregated some of the coding categories (shown as rectangular boxes with heavier lines in Table 3):

- "Learning new physics" and "Learning Andes conventions" imply that the student learned something.

Table 3. Classification of behaviors (columns) and outcomes (rows).

	No hints, much guessing	No hints, little guessing	Asked for all hints, rapidly	Asked for some hints	Referred to example	Asked expert for help	Other	Andes defect	total
Learning new physics	0	2	0	1	0	4	0		7
Learning Andes conventions	5	6	0	30	11	17	3		72
Applying existing knowledge	2	18	0	56	2	0	0		78
Guessed the correction	4	14	0	2	0	1	0		21
Guessing or applying knowledge	1	9	0	5	0	0	0		15
Copied a bottom-out hint	0	0	11	1	2	3	1		18
Deleted the red entry	32	0	0	6	0	0	17	9	64
Miscellaneous correction	2	1	0	3	0	2	0		8
Total	46	50	11	104	15	27	21	9	283

- "Guesses the correction", "Guessing or applying knowledge", "Copying a bottom-out hint" and "Deleting the red entry" imply that the student lost the opportunity to learn something.
- "No hints, much guessing" and "No hints, little guessing" mean that the student never asked for help.
- "Asked for all hints, rapidly", "Asked for some hints" and "Referred to example" mean that the student asked Andes for help.

Table 4 shows such a coarser classification, leaving out the miscellaneous codes. The integers in the cells are the sum of the cells in the corresponding heavy box in Table 3. For instance, the 13 in the cell [Learn, Never ask] is the sum of the four cells in the upper left of Table 3 (0, 2, 5, 6). The cells also show percentages of the row and of the column.

The bottom row of Table 4 indicates that students often asked Andes for help (52%) and seldom asked the experimenter for help (10%), which is reasonable, as that is what they were asked to do. However, even though they were trying to correct a red entry, students often did not ask for help at all (38%). This usually turned out badly for them. Of the episodes where they never asked for help, 65% resulted in them either deleting the red entry, guessing a correction without understanding it, or in some other way losing the opportunity to learn. Only in a few cases (14%) did they manage to learn from their guessing.

Table 4. Coarse classification of behaviors (columns) and outcomes (rows).

	Never ask		Ask Andes		Ask expert		Total
Learn	**13**	17%	**42**	55%	**21**	28%	76
	14%		33%		84%		31%
Apply knowledge	**20**	26%	**58**	74%	**0**	0%	78
	22%		46%		0%		32%
Lost opportunity	**60**	66%	**27**	30%	**4**	4%	91
	65%		21%		16%		37%
Total	93	38%	127	52%	25	10%	245

Now let's consider the 127 episodes where students referred to Andes for help, either by asking for a hint or by referring to its walk-through example. When students faced a red entry, they often (52%) asked Andes for help. Of these episodes, most were beneficial in that the student learned something from the hints (33%) or at least were reminded of something they already knew (46%). Only in a few cases (21%) did students fail to profit from asking Andes for help. Often (11 of 27 episodes; see Table 3) the failure was partly the students' fault because they abused the hints by skipping through them quickly until they got to the bottom out hint, then simply copied it without self-explanation or other apparent thought. In short, the failure rate when students ask Andes for help is 21% whereas the failure rate when they ask for no help is three times larger: 65%. This suggests that students who are inclined not to ask for help should be somehow nudged into asking for more help.

Now for the 25 episodes where the students asked the experimenter for help, the failure rate is 16%, which is not far off from the failure rate for Andes' help (21%). This is consistent with a recent meta-analysis (VanLehn, 2011) that suggests that human tutoring is no more effective than step-based tutoring.

Another way to look at the data is to find out how various outcomes occurred. Let's first examine how the learning occurred. Reading across the top row of Table 4, we see that of the 76 episodes where learning occurred, most (55%) occurred when Andes gave a hint. Of the 78 episodes where students first got a red entry then recalled the correct knowledge and applied it, they asked for a hint on most of them (74%). Once again, this shows that successful learning and practice after receiving a red entry comes mostly when students refer to Andes for help.

If we look across the bottom row, lost opportunities, we see that 66% of them occurred when students did not ask for help. Only a few of them (11 of 27 episodes; see Table 1) were due to hint abuse. Thus, of the two ways to game the system, hint refusal and hint abuse, most lost opportunities came from hint refusal.

Episodes that involved the correction of a red entry were divided into ones where the students struggled (47) and ones where they did not struggle (172). Table 5 shows the classifications in terms of behaviors and outcomes of the episodes classified as struggling.

Table 5. Classification of struggling episodes.

	No hints, much guessing	No hints, little guessing	Asked for all hints, rapidly	Asked for some hints	Referred to example	Asked expert for help	Other	total
Learning new physics	0	1	0	0	0	2	0	3
Learning Andes conventions	5	4	0	3	0	11	3	26
Applying existing knowledge	1	1	0	0	0	0	0	2
Guessed the correction	3	3	0	0	0	1	0	7
Guessing or applying knowledge	0	2	0	0	0	0	0	2
Copying a bottom-out hint	1	0	4	0	0	1	0	6
Miscelaneous correction	1	0	0	0	0	0	0	1
Total	11	11	4	3	0	15	3	47

The bottom row confirms what one would expect intuitively, that struggling episodes are characterized by not asking for help at all or asking for the expert's help. If the student sought help from Andes and this sufficed for correcting the red entry, then the episode was not coded as struggling.

4. Discussion

Perhaps the best overall picture of students' behavior comes from the rightmost column of Table 4. When students' entry is marked incorrect (red), their behavior is evenly divided between learning from this opportunity (31%), recalling knowledge that they seem to have forgotten (32%) and deliberately avoiding the opportunity to learn (37%). The first two cases are a welcome and natural part of the overall learning process; mistakes often provide an opportunity to learn new knowledge or strengthen knowledge that should have applied. Unfortunately, students deliberately avoided the opportunity to learn about a third of the time. Referring to Table 3, we see that they most frequently lost opportunities by deleting the red entry, which amounts to giving up. They also gamed the Andes system by guessing (hint refusal) and by rapidly clicking through the hints in order to reach and copy the bottom-out hint (hint abuse). In short, although students' behavior was productive about 2/3 of the time, their learning could be improved if something could be done to reduce the frequency of lost opportunities.

Toward this goal, the main conclusion seems to be that asking Andes for help is a good idea, and students should do it more often. If students asked for help (127 episodes), then they often learned something new (33%), or were reminded of something they knew already (46%), and the knowledge allowed them to correct the red entry that initiated the episode. On the other hand, if the students tried to work out the correction on their own without asking either Andes or the expert for help (93 episodes), then they often failed to learn or apply knowledge (65%) and often ended up just deleting the red entry.

This is not to say that Andes' help is perfect. It was sometimes abused (11 of 127 episodes), and sometimes resulted in failures to learn or apply knowledge (16 of 127 episodes). Moreover, students sometimes asked the experimenter for help (25 episodes), and some of these represent a failure of Andes to provide adequate help.

There are some limitations of these results. First, these students had already taken physics including a recent midterm on the particular portion of physics addressed in this study, so they seldom learned new physics concepts (7 episodes of 283). What they did learn (72 episodes) was the Andes notation and user interface conventions, such as always using units for dimensional numbers. Second, these data come from the first two hours of Andes usage, so they may not be representative of routine usage of the system. Third, there were only 10 students involved in the study, and they may not be representative of the whole population of potential Andes users. Fourth, we used only one coder instead of multiple coders, so the coding might not be as reliable as it could be.

There are several clear goals for future work. One is to provide proactive hinting to students who are refusing to ask for help even though they keep re-entering an entry incorrectly. This should increase their learning rate during this critical initial encounter

with Andes. A version of Andes that provides such proactive hints has been developed and evaluated. The results, which were positive, will be presented in a future paper.

A second goal would be to recode the data using the categories from a recent study of human physics tutoring (Chi, Roy, & Hausmann, 2008), which found that the measure of tutorial dialogues that most strongly correlated with learning gains was how many times the student responded to the tutor with a substantive comment ($r = 0.605$) versus an non-substantive comment ($r = -0.899$), where a non-substantive comment was a dialogue continuer ("Uh-huh"), merely agreeing with the tutor ("OK."), repeating the tutor's remark, or making an off-task comment. In the context of Andes, after each tutor turn (e.g. coloring an entry red or giving a hint), we would code the student's verbal protocol as substantive vs. non-substantive. It seems likely that we would again see that this distinction correlates strongly with learning. This would extend Chi's hypothesis that tutoring succeeds just to the extent that it gets students thinking hard and making substantive responses. If this correlation was strong, then the distinction could be used as a timely indicator of whether a tutor turn "worked." For instance, if the tutor colors an entry red in a specific context, and the student makes a substantive comment about it, then we have indirect evidence that minimal feedback worked well in that specific context. This kind of indicator could be used to drive a supervised machine learning algorithm that would infer the contexts in which minimal feedback was effective.

A third goal would be to recode the data to answer more specific questions about students' use of hints, such as (1) when and why do students ask for hints? (2) How do they react to hints? For instance, do they self-explain bottom out-hints?

Acknowledgments

References

Aleven, V., & Koedinger, K. R. (2000). Limitations of student control: Do students know when they need help? In G. Gauthier, C. Frasson & K. VanLehn (Eds.), *Intelligent Tutoring Systems: 5th International Conference, ITS 2000* (pp. 292–303). Berlin: Springer.

Baker, R. S., Corbett, A., Koedinger, K. R., Evenson, S., Roll, I., Wagner, A. Z., … Beck, J. (2006). Adapting to when students game an intelligent tutoring system. In *Intelligent Tutoring Systems* (pp. 392–401). Berlin: Springer.

Baker, R. S. J. d., Corbett, A., Koedinger, K. R., & Wagner, A. Z. (2004). Off-task behavior in the cognitive tutor classroom: When students "Game the System". In E. Dykstra-Erickson & M. Tscheligi (Eds.), *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 383–390). New York, NY: ACM.

Baker, R. S. J. d., de Carvalho, A. M. J. A., Raspat, J., Corbett, A., & Koedinger, K. R. (2009). Educational software features that encourage or discourage "gaming the system". In *Proceedings of the 14th International Conference on Artificial Intelligence in Education* (pp. 475–482). Amsterdam: IOS Press.

Baker, R. S. J. d., Walonski, J., Heffernan, N. T., Roll, I., Corbett, A., & Koedinger, K. R. (2008). Why students engage in "gaming the system" behavior in interactive learning environments. *Journal of Interactive Learning Research*, *19*(2), 185–224.

Bloom, B. S. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, *13*, 4–16.

Cade, W. L., Copeland, J. L., Person, N., & D'Mello, S. K. (2008). Dialogue modes in expert tutoring. In B. P. Woolf, E. Aimeur, R. Nkambou & S. P. Lajoie (Eds.), *Intelligent Tutoring Systems: 9th international Conference, ITS2008* (pp. 470–479). Berlin, Germany: Springer.

Chi, M. T. H. (1997). Quantifying qualitative analyses of verbal data: A practical guide. *Journal of the Learning Sciences*, *6*(3), 271–315.

Chi, M. T. H. (2009). Active-Constructive-Interactive: A conceptual framework for differentiating learning activities. *Topics in Cognitive Science*, *1*, 73–105.

Chi, M. T. H., Roy, M., & Hausmann, R. G. M. (2008). Observing tutorial dialogues collaboratively: Insights about human tutoring effectiveness from vicarious learning. *Cognitive Science*, *32*(2), 301–342.

Chi, M. T. H., Siler, S., Jeong, H., Yamauchi, T., & Hausmann, R. G. (2001). Learning from human tutoring. *Cognitive Science*, *25*, 471–533.

Cho, B.-I., Michael, J. A., Rovick, A. A., & Evens, M. W. (2000). An analysis of multiple tutoring protocols. In G. Gauthier, C. Frasson & K. VanLehn (Eds.), *Intelligent Tutoring Systems: 5th International Conference* (pp. 212–221). Berlin: Springer.

Corbett, A. (2001). Cognitive computer tutors: Solving the two-sigma problem. In M. Bauer, P. J. Gmytrasiewicz & J. Vassileva (Eds.), *User Modeling: Proceedings of the Eighth International Conference* (pp. 137–147). Berlin: Springer-Verlag.

Core, M. G., Moore, J. D., & Zinn, C. (2003). The role of initiative in tutorial dialogue. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)* (Vol. Morristown, NJ, pp. 67–74): Association of Computational Linguistics.

Evens, M., & Michael, J. (2006). *One-on-one tutoring by humans and machines*. Mahwah, NJ: Erlbaum.

Fox, B. A. (1991). Cognitive and interactional aspects of correction in tutoring. In P. Goodyear (Ed.), *Teaching knowledge and intelligent tutoring* (pp. 149–172). Norwood, NJ: Ablex.

Fox, B. A. (1993). *The human tutorial dialogue project: Issues in the design of instructional systems*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Frederiksen, N., Donin, J., & Roy, M. (2000). *Human tutoring as a model for computer tutors: Studying human tutoring from a cognitive perspective.* Paper presented at the Modeling human teaching tactics and strategies: Workshop W1 of ITS2000, Montreal, Canada.

Graesser, A. C., Person, N., & Magliano, J. (1995). Collaborative dialog patterns in naturalistic one-on-one tutoring. *Applied Cognitive Psychology*, *9*, 359–387.

Graesser, A. C., VanLehn, K., Rose, C. P., Jordan, P., & Harter, D. (2001). Intelligent tutoring systems with conversational dialogue. *AI Magazine*, *22*(4), 39–41.

Hastings, P., Arnott-Hill, E., & Allbritton, D. (2010). Squeezing out gaming behavior in a dialog-based ITS. In V. Aleven, H. Kay & J. Mostow (Eds.), *Intelligent Tutoring Systems 2010* (pp. 204–213). Berlin: Springer-Verlag.

Hume, G., Michael, J., Rovick, A., & Evens, M. (1996). Hinting as a tactic in one-on-one tutoring. *Journal of the Learning Sciences*, *5*(1), 23–49.

Katz, S., Allbritton, D., & Connelly, J. (2003). Going beyond the problem given: How human tutors use post-solution discussions to support transfer. *International Journal of Artificial Intelligence in Education*, *13*, 79–116.

McArthur, D., Stasz, C., & Zmuidzinas, M. (1990). Tutoring techniques in algebra. *Cognition and Instruction*, *7*(3), 197–244.

Merrill, D. C., Reiser, B. J., Ranney, M., & Trafton, J. G. (1992). Effective tutoring techniques: A comparison of human tutors and intelligent tutoring systems. *The Journal of the Learning Sciences*, *2*(3), 277–306.

Merrill, D. C., Reiser, B. J., Merrill, S. K., & Landes, S. (1995). Tutoring: Guided learning by doing. *Cognition and Instruction*, *13*(3), 315–372.

Muldner, K., Burleson, W., van de Sande, B., & VanLehn, K. (2010). An analysis of gaming behaviors in an intelligent tutoring system. In V. Aleven, J. Kay & J. Mostow (Eds.), *Intelligent Tutoring Systems: 10th International Conference, ITS 2010* (pp. 184–193). Heidelberg, Germany: Springer.

Muldner, K., Burleson, W., van de Sande, B., & VanLehn, K. (2011). An analysis of students' gaming behaviors in an intelligent tutoring system: Predictors and impacts. *User Modeling and User-Adapted Interaction*, *21*(1-2), 99–135.

Muldner, K., & Conati, C. (2010). Scaffolding meta-cognitive skills for effective analogical problem solving via tailored example selection. *International Journal of Artificial Intelligence and Education*, *20*, 99–136.

Murray, R. C., & VanLehn, K. (2005). Effects of dissuading unnecessary help requests while providing proactive help. In G. I. McCalla, C.-K. Looi, B. Bredeweg & J. A. Breuker (Eds.), *Proceedings of Artificial Intelligence in Education* (pp. 887–889). Amsterdam: IOS Press.

Ohlsson, S., Di Eugenio, B., Chow, B., Fossati, D., Lu, X., & Kershaw, T. C. (2007). Beyond the code-and-count analysis of tutoring dialogues. In R. Luckin, K. R. Koedinger & J. Greer (Eds.), *Artificial Intelligence in Education* (pp. 349–356). Amsterdam: IOS.

Ringenberg, M., & VanLehn, K. (2006). Scaffolding problem solving with annotated, worked-out examples to promote deep learning. In K. Ashley & M. Ikeda (Eds.), *Intelligent Tutoring Systems: 8th International Conference, ITS2006* (pp. 114–123). Amsterdam: IOS Press.

Shih, B., Koedinger, K. R., & Scheines, R. (2008). A response time model for bottom-out hints as worked examples. In C. Romero, S. Ventura, M. Pechenizkiy & R. S. J. d. Baker (Eds.), *Handbook of Educational Data Mining* (pp. 201–211). Boca Raton, FL: Taylor & Francis.

VanLehn, K. (1999). Rule learning events in the acquisition of a complex skill: An evaluation of Cascade. *Journal of the Learning Sciences*, *8*(2), 179–221.

VanLehn, K. (2006). The behavior of tutoring systems. *International Journal of Artificial Intelligence and Education*, *16*, 227–265.

VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems and other tutoring systems. *Educational Psychologist*, *46*(4), 197–221.

VanLehn, K., Graesser, A. C., Jackson, G. T., Jordan, P., Olney, A., & Rose, C. P. (2007). When are tutorial dialogues more effective than reading? *Cognitive Science*, *31*(1), 3–62.

VanLehn, K., Lynch, C., Schultz, K., Shapiro, J. A., Shelby, R. H., Taylor, L., … Wintersgill, M. C. (2005). The Andes physics tutoring system: Lessons learned. *International Journal of Artificial Intelligence and Education*, *15*(3), 147–204.

VanLehn, K., Siler, S., Murray, C., Yamauchi, T., & Baggett, W. B. (2003). Human tutoring: Why do only some events cause learning? *Cognition and Instruction*, *21*(3), 209–249.

Woolf, B. P. (2009). *Building intelligent interactive tutors*. Burlington, MA: Morgan Kaufman.