Chapter 21
The Andes Physics Tutoring System: An Experiment in Freedom

Kurt VanLehn, Brett van de Sande, Robert Shelby, and Sophia Gershman

School of Computing, Information, Informatics and Decision Systems Engineering,
Arizona State University, 699 S. Mill Avenue, Tempe, AZ 85287
{kurt.vanlehn,bvsd}@asu.edu

Abstract. The Andes physics tutoring system is an experiment in student freedom. It allows students to solve a physics problem in virtually any legal way. This means that Andes must recognize an extremely large number of possible steps occurring in an extraordinarily large number of possible orders. Such freedom raises several research questions. (1) How can Andes solve the technical challenge of understanding student's behavior in such a wide-open context? (2) How can Andes give pedagogically useful help and guidance? In particular, how can it guide students who are floundering without curtailing the freedom of students who are not floundering? (3) Will Andes be effective in getting students in real classrooms to learn physics? (4) What does it take to scale up Andes and disseminate it widely? The Andes project, which began in the mid 1990's, has achieved workable solutions to the first three goals: Andes can understand student behavior; It provides pedagogical help similar to that of human experts; Most importantly, Andes causes large, reliable learning gains compared to control classes taught with convention, paper-based instruction. This chapter summarizes the first three results and discusses our progress on the fourth goal, scale-up.

21.1 Introduction

Although problem solving freedom may seem a desirable goal in its own right, it was adopted as the driving goal for the Andes system in the belief that allowing freeform problem solving would facilitate the fourth goal, scale-up. Given a particular physics problem, different instructors will endorse different solutions, and for good reasons. For instance, some instructors prefer mathematically concise solutions in the belief that this makes the physics stand out. Other instructors prefer verbose solutions that make explicit every physics principle's application in the belief that this makes the physics stand out. It may be that both instructors are completely correct, because they are working with different students or they are preparing their students for different subsequent courses. Andes should not take a stand on this issue, but instead should support all instructors' pedagogical

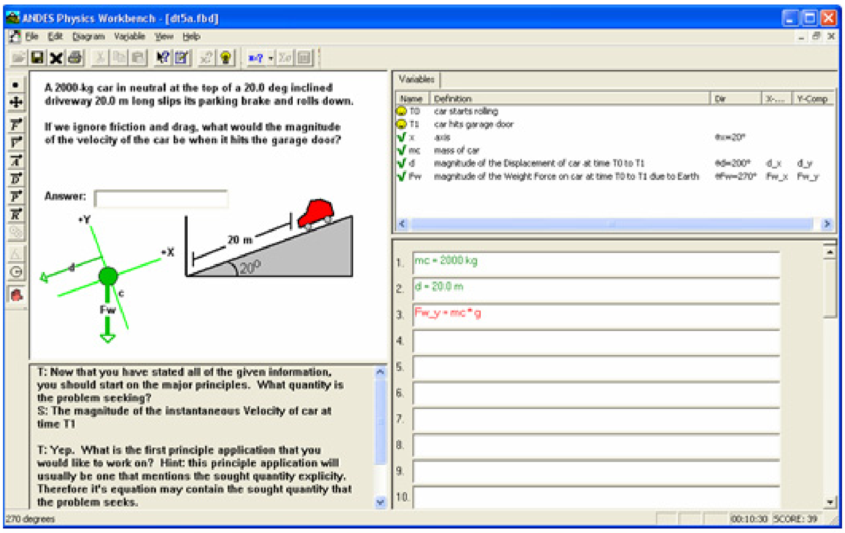
R. Nkambou et al. (Eds.): Advances in Intelligent Tutoring Systems, SCI 308, pp. 421–443.
springerlink.com

practices equally. To put it succinctly, pedagogical equality motivated problem solving freedom.

The plan for this chapter is to first describe how students solve physics problems on Andes, and highlight the freedom they have by comparing Andes' user interface to the user interfaces of other well-known intelligent tutoring systems (ITS). The second section explains how Andes solved the technological problems of scaffolding students despite the freedom they enjoyed. A third section describes evaluations of Andes in the physics classes of the United States Naval Academy. The final section reports progress on scaling up and disseminating Andes.

21.2 The User Interface and Behavior of Andes

Andes' user interface is intended to be as much like pencil and paper as possible. The original version of the Andes user interface is shown in Figure 21.1; a more recently developed one will be discussed later. Students read the problem (top of the upper left window), draw vectors and coordinate axes (bottom of the upper left window), define variables (upper right window) and enter equations (lower right window). These are actions that they do when solving physics problems with pencil and paper.


ANDES Physics Workbench - [dt5a.fbd]
File Edit Diagram Variable View Help
+Y
+X
d
c
Fw
20 m
20°
Variables
270 degrees
00:10:30
SCORE: 39

A 2000-kg car in neutral at the top of a 20.0 deg inclined driveway 20.0 m long slips its parking brake and rolls down.

If we ignore friction and drag, what would the magnitude of the velocity of the car be when it hits the garage door?

Answer:

Name	Definition	Dir	X-...	Y-Comp
T0	car starts rolling			
T1	car hits garage door			
✓ x	axis	θx=20°		
✓ mc	mass of car			
✓ d	magnitude of the Displacement of car at time T0 to T1	θd=200°	d_x	d_y
✓ Fw	magnitude of the Weight Force on car at time T0 to T1 due to Earth	θFw=270°	Fw_x	Fw_y

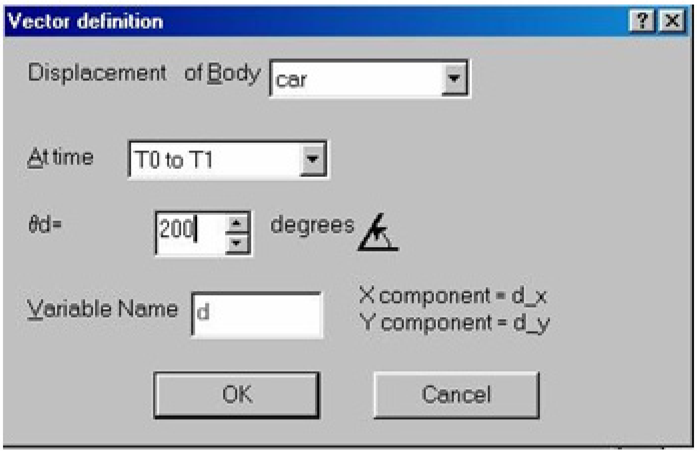
1. mc = 2000 kg
2. d = 20.0 m
3. Fw_y = mc*g
4.
5.
6.
7.
8.
9.
10.

T: Now that you have stated all of the given information, you should start on the major principles. What quantity is the problem seeking?
S: The magnitude of the instantaneous Velocity of car at time T1

T: Yep. What is the first principle application that you would like to work on? Hint: this principle application will usually be one that mentions the sought quantity explicity. Therefore it's equation may contain the sought quantity that the problem seeks.

Fig. 21.1 The Andes2 user interface

Although similar to paper and pencil in many ways, Andes' user interface does differ several important ways. First, as soon as an action is done, Andes gives immediate feedback. Entries are colored green if they are correct and red if they are incorrect. This binary indicator is called flag feedback (Anderson et al. 1995). In Figure 21.1, all the entries are green except for equation 3, which is red.



Vector definition

Displacement of Body car

At time T0 to T1

θd= 200 degrees

Variable Name d

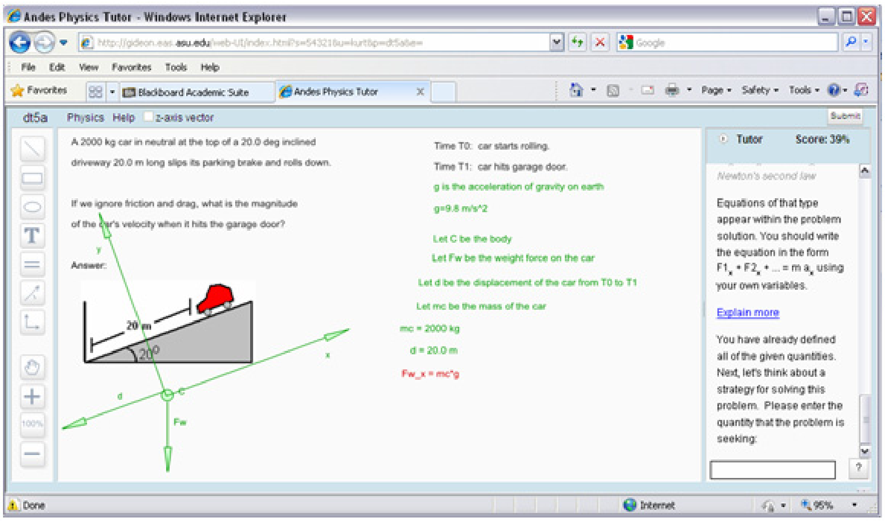
X component = d_x
Y component = d_y

Fig. 21.2 A form for defining a vector

Second, variables must be defined before they can be used in equations. There are two kinds of quantities: vectors and scalars. Vectors are defined by clicking on the symbol for the appropriate vector variable on the tool bar on the left edge of Figure 21.1, then drawing an arrow using the mouse, then filling out a pop-up form like the one in Figure 21.2. Scalar variables are defined by filling out a similar form. Filling out these forms forces students to precisely define the semantics of quantities. Paper does not enforce such precision, so students can easily confuse their variables. For instance, students working on paper often use “v” to stand for all the velocities in a problem. Our collaborating instructors at the United States Naval Academy (USNA) strongly believe that requiring students to define quantities before using them in equations accelerates their learning even though it does restrict their freedom somewhat.

Another violation of freedom is the forms that students fill out when defining quantities. Users must fill in all the blanks before they can close the form. Sophisticated users consider some of the information so obvious that they don’t want to fill those blanks in. For instance, when there are only two time points in a problem, there can be only one time interval in a problem, so “duration” must refer to the length of that time interval and “displacement” must refer to the change in location over that time interval. When sophisticated users write such definitions on paper, they often do not use temporal subscripts. They include subscripts only when needed for disambiguation. This was a constant source of irritation to some (but not all) instructors using Andes.

The newest version of the Andes user interface, which is shown in Figure 21.3, does not use forms for defining quantities. Instead, users open a textbox and type their definition into it. Andes matches the user’s definition to all definitions possible in this problem, picks the closest match, and confirms it with the user. The text then appears on the interface inside the textbox dragged out by the user. Thus, Andes will accept “Let d be the displacement” and does not insist on the



A 2000 kg car in neutral at the top of a 20.0 deg inclined driveway 20.0 m long slips its parking brake and rolls down.

If we ignore friction and drag, what is the magnitude of the car's velocity when it hits the garage door?

Answer:

Time T0: car starts rolling.

Time T1: car hits garage door.

g is the acceleration of gravity on earth

g=9.8 m/s^2

Let C be the body

Let Fw be the weight force on the car

Let d be the displacement of the car from T0 to T1

Let mc be the mass of the car

mc = 2000 kg

d = 20.0 m

Fw_x = mc*g

Equations of that type appear within the problem solution. You should write the equation in the form $F1_x + F2_x + ... = m\ a_x$ using your own variables.

You have already defined all of the given quantities. Next, let's think about a strategy for solving this problem. Please enter the quantity that the problem is seeking:

Fig. 21.3 The Andes3 user interface

more complete definition, "Let d be the displacement of the car from T0 to T1." On the other hand, "Let v be the velocity" is red flagged because it is ambiguous between the instantaneous velocity at T0, the instantaneous velocity at T1 and the average velocity between T0 and T1. This version of Andes (called Andes3) is in beta testing as this chapter is being written. We call it a *freeform* user interface both because it has no forms and because it puts little constraint on the steps and the ordering of steps (except that quantities must be defined before appearing in equations). The new freeform user interface is even closer to paper and pencil.

Andes includes a mathematics package. Students indicate which variable they want to solve for, then Andes tries to solve the system of equations that the student has entered for numerical result for the variable. If Andes succeeds, it enters an equation of the form <variable> = <value>. Although many students routinely use powerful hand calculators and computer-based mathematics packages, such usage requires copying the equations from Andes to their system and back. Andes eliminates this tedious and error-prone copying process. This is one reason that Andes is popular with students. Nonetheless, instructors can turn this feature off.

Andes provides three kinds of help:

Andes pops up an error messages whenever the error is likely to be a slip. That is, the error is probably due to lack of attention rather than lack of knowledge (Norman 1981). Typical slips include using an undefined variable in an equation (which is usually caused by a typographical error) or leaving off the units of a dimensional number. When an error is not recognized as a slip, Andes colors the entry red.

Students can request help on a red entry by selecting it and clicking on a help button. Since the student is essentially asking, "what's wrong with that?" we call this *What's Wrong Help*.

If students are not sure what to do next, they can click on a button that will give them a hint. This is called *Next Step Help*.

Notice the students' freedom. Although Andes takes the initiative when it detects a slip, Andes only gives substantive hints only when students ask for them.

What's Wrong Help and Next Step Help generate a hint sequence. The first time the student asks for hints, the first hint in the sequence is printed in the lower left window (Fig. 21.1) or the right window (Fig. 21.3). The student can then ask for another hint, make a new entry or fix an existing red entry. If the student asks for another hint without making any correct entries, then the second hint in the sequence is printed. Most hint sequences have three hints. As an illustration, suppose a student who is solving Figure 21.1 has asked for What's Wrong Help on the incorrect equation *Fw_x = -Fs*cos(20 deg)*. These are the three hints that Andes gives:

Check your trigonometry.

If you are trying to calculate the component of a vector along an axis, here is a general formula that will always work: Let θV be the angle as you move counterclockwise from the horizontal to the vector. Let θx be the rotation of the x-axis from the horizontal. Then: V_x = V*cos(θV-θx) and V_y = V*sin(θV-θx).

Replace cos(20 deg) with sin(20 deg).

This three-hint sequence is typical of many hint sequences. It is composed of a *pointing* hint, a *teaching* hint and a *bottom-out* hint. The pointing hint, "Check your trigonometry," directs the students' attention to the location of the error. If the student knows the appropriate knowledge and the mistake is due to carelessness, then the student should be able to pinpoint and correct the error given such a hint (Hume et al. 1996; Merrill et al. 1992).

The teaching hint, "If you are trying to calculate…," states the relevant piece of knowledge. We try to keep these hints as short as possible, because students tend not to read long hints (Anderson et al. 1995; Nicaud et al. 1999). In other work, we have tried replacing the teaching hints with either multimedia (Albacete and VanLehn,2000a, 2000b) or natural language dialogues (Rose et al. 2002). These more elaborate teaching hints significantly increased learning in laboratory settings, but have not been tried in the field.

The bottom-out hint, "Replace cos(20 deg) with sin(20 deg)," tells the student exactly what to do. Students often abuse these bottom out hints, so Andes uses a scoring function (see below) to discourage such "gaming the system" (Baker et al. 2009).

This completes the description of the Andes' user interface. Except for one major restriction (quantities must be defined before using them in equations), students have the freedom to do any step in any order, and to ask for hints at any time. Although Andes does give flag feedback (i.e., red/green) on every step without being asked to do so, students surveys indicate that they like this. If given a choice, they probably would ask for such feedback after every step. In short, with a few exceptions, the Andes user interface gives the user complete freedom to develop solutions.

In contrast to Andes, many ITS constrain student's freedom. For this discussion, let us focus on step-based tutoring systems and ignore answer-based tutoring systems (VanLehn 2006). The difference is that answer-based tutoring systems given feedback and hints on the answer as a whole, whereas step-based tutoring systems decompose the student's entries into steps and give feedback and hints on each step. Most ITS are step-based tutors; Andes is a step-based tutoring system, where a step is a student entry, such as an equation or a quantity definition.

Anyway, many step-based ITS have students enter each step into a specific blank in a form. Only certain steps are allowed in that blank, and the blanks are often labeled. Thus, each blanks represents (reifies) a goal in solving the problem. For instance, the user interface for the SQL Tutor (Mitrovic 2003; Mitrovic and Ohlsson 1999) has blanks for each component of an SQL query. The blanks are labeled (e.g., "Select", "From" or "Where"). Thus, the overall goal of formulating an SQL query is decomposed into several subgoals, and each subgoal is made explicit (reified) as a blank plus its label. This goal reification is part of the scaffolding provided by the SQL tutor and probably contributes to its effectiveness. However, it does reduce the student's freedom. If the SQL tutor's user interface consisted merely of a large text box into which student's entered a whole query, then students would have more freedom.

As a contrasting example, suppose a form-based interface were used for physics problems. Its blanks would have labels such as "Newton's second law applied on the x-axis:" or "Definition of the acceleration of the bowling ball:"

It is an empirical question whether the form-based interface or the freeform interface is more effective, and for which students. Novices may learn faster with a form-based interface whereas intermediates may learn faster with a freeform interface (Singley, 1990). Perhaps transfer from freeform interfaces to paper might be better than transfer from form-based interfaces to paper. The point is merely that the Andes freeform user interface is atypical. Most step-based ITS use form-based interfaces even though they could use a freeform one, and Andes could use a form-based interface even though it uses a freeform one.

21.3 Scaffolding Learning when Students Have Freedom

When a user interface gives students freedom to do any step, students can flounder and they can develop bad problem solving habits. The Andes project developed methods for dealing with both problems and thus scaffolding the students' learning even in the context of a form-free user interface.

When students flounder, most eventually ask for Next Step Help. Two versions of Next Step Help have been developed and evaluated. The first version of Andes (called Andes1) attempted to recognize the student's plan for solving the problem and hint the next step in that plan (Gertner et al. 1998; Gertner and VanLehn 2000). A Bayesian net was used for both plan recognition and next step selection. This approach was used in the 1999 and 2000 experiments. After the 2000 experiment, we evaluated it by randomly selecting episodes where students asked for Next Step Help. For each episode, the student's screen was printed just prior to Andes' first hint, the hard copies were given to 3 physics instructors, and the instructor wrote the sequence of hints they would give at this point. Although the instructors sometimes disagreed on what step to hint next, they did agree in 21 of the 40 episodes. Unfortunately, Andes tended to disagree with the humans. Its hint sequence agreed with the consensual one in only 3 of 21 cases. Thus, Andes1's Next Step Help was not consistent with expert human help.

The problem, in the opinion of instructors, was that students often had no coherent plan for solving the problem. They had generated a few correct entries, often with significant help from Andes, but there appeared to be no pattern in their selection of steps. For these students, the instructors wrote hint sequences that identified the major principle and the first step toward applying that principle. Instead of trying to recognize the student's plan, which is what Andes1 tried to do, the expert instructors hinted the first step in their own plan for solving the problem.

The current version of Next Step Help replicated the instructors' hint sequences. It was based on the assumption that if students are lost and asking for Next Step Help, then they probably have no coherent plan, so they should get a hint on the first relevant step in the instructors' plan for solving the problem. Thus, unless the students have already been introduced to the instructor's plan, Andes walks them through the goal-subgoal-subsubgoal etc chain that leads down to the to-be-hinted step. Figure 21.1 shows a part of such a discussion.

A second problem that can develop with freeform user interfaces is that students can develop bad problem solving habits. Some bad habits are specific to physics, such as plugging in numbers prematurely. (Most instructors prefer that students first generate all the equations needed to solve a problem, and then plug in numbers in order to solve the equations.) Other bad habits involve misuse of the tutoring system itself. For instance, students may "game the system" by clicking rapidly on the hint request button until they get the bottom out hint (e.g. Aleven and Koedinger 2000; Baker et al. 2006).

In order to maintain freedom while discouraging bad problem solving practices, Andes computes and displays a score. Students get points for entering steps correctly without help, and they lose points for gaming and for bad physics practices. The overall score on a problem is continually displayed in the lower right corner. Instructors who use Andes in their classes, and thus can observe students' behavior, report that some students pay a lot of attention to these scores, perhaps even too much.

Note that these scores are not an estimate of the student's mastery of physics. Although Andes1 maintained such a student model, the USNA instructors did not use it for grading. Moreover, they wanted all students to do the same homework problems, so Andes was not allowed to implement mastery learning (Bloom 1984) or macro-adaptation (Shute 1993) by selecting different physics problems for different students. Because there was no use for the model's assessments of student mastery, subsequent versions of Andes did not include such a student model.

21.4 Design and Implementation

This section describes how Andes handles the technical issue of allowing students maximal freedom in entering steps, while still being able to recognize steps and scaffold learning. This section is not an exhaustive description of the design and implementation, as that appears in (VanLehn et al. 2005).

The key to coping with unconstrained physics problem solving is that for each problem, every possible correct step is a mathematical combination of primitive steps. Before the Andes system is distributed, its problem solver solves every problem and stores the set of primitive solution steps in a data structure called a solution graph. There is one solution graph per problem, and it represents all possible solutions allowed by the physics knowledge base in terms of the finest grained steps possible. A typical problem may have hundreds of primitive steps, most of which are equations.

For each problem, Andes also precomputes the problem's *solution point*. For each variable in the solution, the solution point states its value, and there can be hundreds of such variable-value pairs. If the problem has symbolic rather than a numerical givens, then Andes generates "ugly" values for the symbolic givens and computes the solution point in terms of them. For instance, if the problem begins, "A car of mass *m* slides down a frictionless driveway inclined at θ degrees from the horizontal…" then Andes might substitute 142.56721 kg for *m*, and 21.8762° for θ. These values are never visible to the student.

Given the solution graph and the solution point, Andes needs to be able to give immediate flag feedback, What's Wrong Help and Next Step Help. The following subsections describe each.

21.4.1 Implementing Immediate Feedback

This section describes how Andes implements immediate feedback. When the student makes an entry, Andes' immediate feedback consists of either coloring it green, coloring it red or popping up an error message such as "Undefined variable."

From an implementation perspective, there are two types of entries that students make when solving a problem: equations and non-equations. Non-equation entries include drawing vectors, drawing coordinate axes and defining scalar variables. A non-equation entry is compared to the non-equation entries in the solution graph. If an exact match is found, the student's entry is colored green; otherwise it is colored red. Thus, a non-equation entry is considered correct if and only if it is a primitive step in at least one solution to the problem.

This simple matching technique won't work for equation entries, because students can enter an equation that is an algebraic combination of primitive equations. Such a composite equation will not be in the solution graph, but it is nonetheless correct. Moreover, an infinite number of correct equations can be generated algebraically from even a single equation (e.g., add 3.1 to both sides).

Thus, there are too many correct equations to precompute and store. This is the price of allowing students' freedom to enter composite equations.

Andes checks equations for correctness using a technique called *color by numbers* (Shapiro 2005). The solution point is substituted into the student's equation. This substitution produces an arithmetic equation. That is, the equation has all numbers and no variables. If the equation balances, with some allowance for round-off errors, then the student's entry is colored green; and red otherwise.

Color-by-numbers is similar to a technique used by tutoring systems ever since Plato (Kane and Sherwood 1980). However, the older technique applies to algebraic *formulas* whereas Andes' technique applies to algebraic *equations*. For instance, suppose a tutoring system asks a fill-in-the-blank question such as "*v1_x* = __________" or "The x-component of the initial velocity is __________" and expects an algebraic expression as the answer. It can check the correctness of the student's entry by substituting numbers for variables and simplifying the resulting arithmetic formula. If simplification yields the expected number, then the student's answer is correct. Color-by-numbers essentially does this to both sides of the equation.

Although it was always clear that the solution point could be substituted into an equation and the resulting arithmetic equation could checked for balance, it was not clear what such a check really meant. Joel Shapiro proved that this check was equivalent to finding a algebraic derivation from the primitive equations. Thus, color-by-numbers can be used to replace search-based feedback used by earlier systems (Brna and Caiger 1992; Yibin and Jinxiang 1992) and by Andes1. Andes2 may be the first system to apply the color-by-numbers technique to equations instead of expressions.

21.4.2 Implementing What's Wrong Help

When a red entry is selected and the student clicks on the What's Wrong Help button, Andes applies a large set of error handlers to the entry. Each error handler takes the student's entry as input. If it recognizes the error, it returns a hint sequence and a priority. If several error handlers recognize errors in an incorrect entry, Andes chooses the hint sequence with the highest priority.

Many ITS recognize errors by matching the student's incorrect step to each of the possible correct steps that the student should have made. The match will be partial, because the student's step is incorrect. When the ITS has finished partially matching the student's step to all the correct steps, it selects the best partial match and use it to present a hint sequence. For form-based user interfaces, there are usually only a small number of correct steps that can be entered into a blank in the form, so matching the student's step against each of them is feasible. However, with Andes' freeform approach, there are far too many possible correct steps to match against. Moreover, for equations, the student could be trying to enter a composite equation instead of a primitive one. If Andes were to use the conventional matching approach, it would have to generate all correct composite equations and match them to the student's step. This is clearly infeasible. Thus, Andes had to use a different approach.

Andes recognizes errors by making edits to the student's entry. Each error handler makes different kinds of edits. If the edit generates a correct entry, then the error handler constructs an appropriate hint sequence that refers to the student's entry. For instance, suppose a student who is solving the problem in Figure 21.1 defines a zero-length velocity vector and asserts that it is instantaneous velocity of the car at T1. The velocity is actually non-zero, so this is an error and it turns red. One edit is to change the time specification, so that the vector stands for the instantaneous velocity of the car at T0. Another edit is to change the qualitative length, so that the vector is non-zero. The second edit gets a higher priority, so Andes selects this hint sequence:

Is the car at rest at T1?

Since the car is not at rest at T1, the velocity vector needs to be non-zero.

For equations, error handlers edit the equation then call color-by-numbers on the edited equation. For instance, suppose a student solving the problem of Figure 21.1 enters the incorrect equation *Fw_x =-Fw*cos(20 deg).* An error handler notes that there is a cosine in the equation, edits the equation to be *Fw_x = -Fw*sin(20 deg)* and submits it to color-by-numbers, which indicates that the edited equation is correct. The error handler returns a high priority hint sequence that starts with "Check your trigonometry."

Often, different error handlers make the same edit, but they have different applicability conditions. For instance, sign errors are common in equations, so there is a general sign-error handler whose hint sequence starts out with "Check your signs." This error handler has a low priority. There is a higher priority error handler that checks for a special kind of sign error. If the term with the sign error is the magnitude of a vector pointing opposite one of the coordinate axes, the hint sequence is (problem-specific text is substituted for the bracketed text):

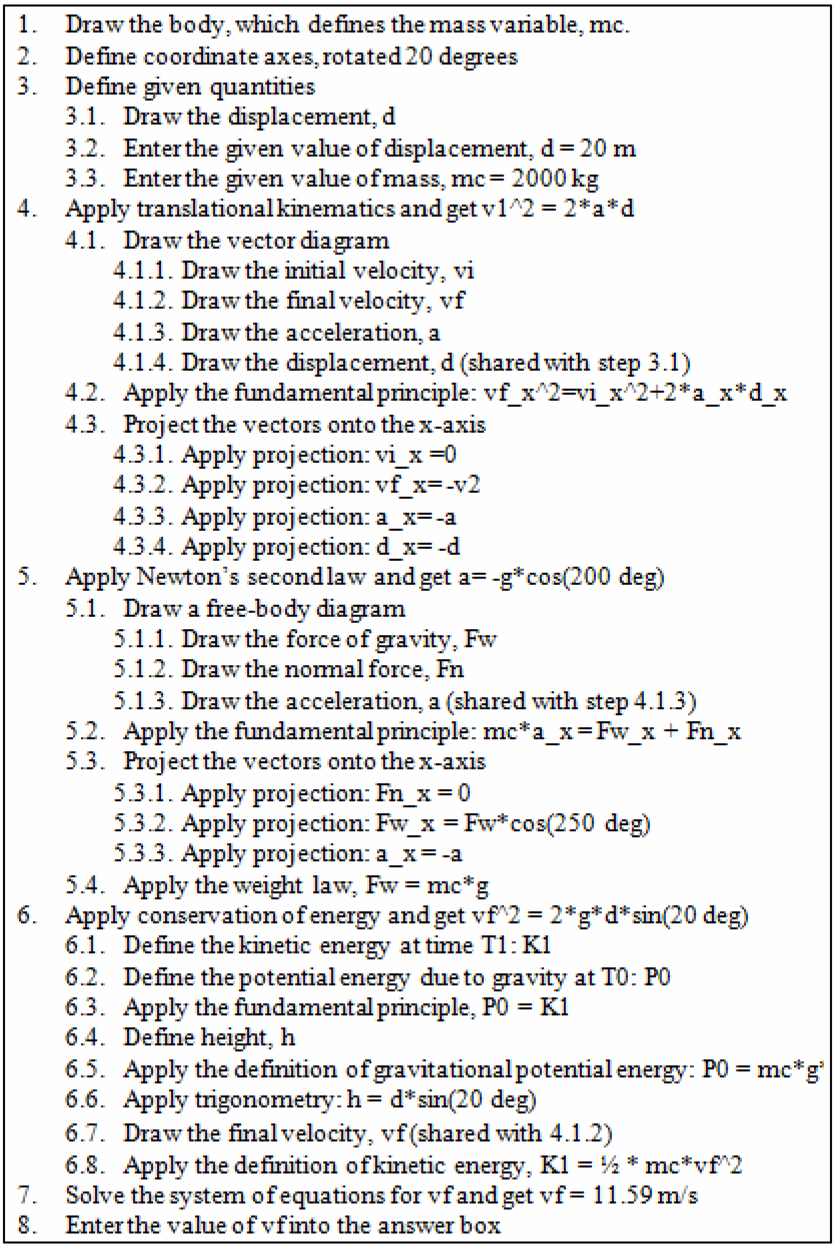
Think about the direction of the <vector>.

Perhaps you are confusing the MAGNITUDE of the <vector> with its COMPONENT along the <x or y> axis. Because the vector is parallel to the <x or y> axis but in the negative direction, the projection equation is <equation>.

Because the vector is parallel to the <x or y> axis and in the negative direction, replace <magnitude variable> with either -<magnitude variable> or <component variable>.

21.4.3 Next Step Help

Some problems have multiple solutions. For instance, the problem in Figure 21.1 can be solved either with conservation of energy or with Newton's second law. These solutions can share parts, such as defining the given quantities. Thus, a solution graph represents plans for every solution and how they share their parts. For example, Figure 21.4 is part of the information in the solution graph for the problem of Figure 21.1. Although not shown in the figure, there is a branch from step 3 to both step 6 and step 4. This represents that there are alternative solutions. These branches converge later; steps 5 and 6 both point to step 7.



1. Draw the body, which defines the mass variable, mc.
2. Define coordinate axes, rotated 20 degrees
3. Define given quantities
    3.1. Draw the displacement, d
    3.2. Enter the given value of displacement, d = 20 m
    3.3. Enter the given value of mass, mc = 2000 kg
4. Apply translational kinematics and get v1^2 = 2*a*d
    4.1. Draw the vector diagram
        4.1.1. Draw the initial velocity, vi
        4.1.2. Draw the final velocity, vf
        4.1.3. Draw the acceleration, a
        4.1.4. Draw the displacement, d (shared with step 3.1)
    4.2. Apply the fundamental principle: vf_x^2=vi_x^2+2*a_x*d_x
    4.3. Project the vectors onto the x-axis
        4.3.1. Apply projection: vi_x =0
        4.3.2. Apply projection: vf_x=-v2
        4.3.3. Apply projection: a_x=-a
        4.3.4. Apply projection: d_x= -d
5. Apply Newton's second law and get a= -g*cos(200 deg)
    5.1. Draw a free-body diagram
        5.1.1. Draw the force of gravity, Fw
        5.1.2. Draw the normal force, Fn
        5.1.3. Draw the acceleration, a (shared with step 4.1.3)
    5.2. Apply the fundamental principle: mc*a_x = Fw_x + Fn_x
    5.3. Project the vectors onto the x-axis
        5.3.1. Apply projection: Fn_x = 0
        5.3.2. Apply projection: Fw_x = Fw*cos(250 deg)
        5.3.3. Apply projection: a_x = -a
    5.4. Apply the weight law, Fw = mc*g
6. Apply conservation of energy and get vf^2 = 2*g*d*sin(20 deg)
    6.1. Define the kinetic energy at time T1: K1
    6.2. Define the potential energy due to gravity at T0: P0
    6.3. Apply the fundamental principle, P0 = K1
    6.4. Define height, h
    6.5. Apply the definition of gravitational potential energy: P0 = mc*g
    6.6. Apply trigonometry: h = d*sin(20 deg)
    6.7. Draw the final velocity, vf (shared with 4.1.2)
    6.8. Apply the definition of kinetic energy, K1 = ½ * mc*vf^2
7. Solve the system of equations for vf and get vf = 11.59 m/s
8. Enter the value of vf into the answer box

Fig. 21.4 The solution graph for the problem of Figure 21.1

The basic job of Next Step Help is to direct the student to some overall problem solving strategy, then guide them through the steps associated with that strategy, taking into account the work the student has already finished. At any time during this process, the student may abort the help and continue solving the problem. The student is free to ignore the advice given and pursue their own problem solving strategy.

Andes first makes sure that the student has "set up" the problem by defining given quantities, bodies, coordinate axes, etc. For instance, if the student has done steps 1 and 3 of Figure 21.4, then Andes will hint step 2.

If the student has already set up the problem, then Andes will help the student select a major principle to apply. For instance, suppose the student has done steps 1, 2, 3 and 6.4, then asks for help. Andes asks, "What quantity is the problem seeking" then "What major principle should you apply to find it?" If the student selects Conservation of Energy, then Andes hints step 6.1, which is the first step of that solution. On the other hand, if the student and Andes have already agreed on a major principle, then Andes will just remind the student of it ("Why don't you continue applying Conservation of Energy to the car.") then hint the first step that has not been done.

Once Andes has selected a step to hint, it generates a hint sequence. It typically consisted of a pointing hint, a teaching hint and a bottom out hint, as illustrated earlier.

In order to make this policy work, Andes must map the student's entries onto steps in the solution graph so that it can mark those steps "done." This is easy for non-equation entries, but it is extremely difficult for equations. For instance, if the student enters *Fw_x = -m*g*sin(20 deg)*, then Andes must somehow figure out that this maps to step 5.3.2 (writing *Fw_x = Fw*cos(250 deg)*) and step 5.4 (writing *Fw = m*g*). Andes1 solved this problem by precomputing all possible algebraic combinations of the steps, but this became infeasible as problems became more complex and numerous. For a long time, this appeared an insurmountable problem.

Fortunately, a solution was found (Shapiro 2005). The algorithm is called *indy check* because its main step is to check the independence of a set of multidimensional vectors. The vectors are the gradients of the solution graph equations and the student's equation. The algorithm computes the gradient of an equation by taking the partial derivative of each variable in the equation, evaluating the resulting expressions at the solution point, and using the resulting numbers as components of the gradient. In order to find a set of solution graph equations that can be combined to form the student's equation, the algorithm finds a set of gradients that can be combined via a weighted sum to form the gradient of the student's equation. The algorithm outputs all such sets.

21.5 Evaluations of Andes at the United States Naval Academy

Andes was evaluated in the USNA's introductory physics class every fall semester from 1999 to 2003. This section describes how the 5 evaluations were conducted and their results.

21.5.1 The Evaluation Method

Andes was used as part of the normal USNA physics course. The course had multiple sections, each taught by a different instructor. Students in all sections took the same final exam and used the same textbook but different instructors assigned different homework problems and gave different hour exams, where hour exams were in-class, hour-long exams given approximately monthly. (Hour exams are often called "quizzes" or "midterms" at other schools.) In sections taught by Andes developers (Profs. Robert Shelby, Donald Treacy and Mary Wintersgill), students did their homework on Andes.

Andes was used as part of the normal USNA physics course. The course had multiple sections, each taught by a different instructor. Students in all sections took the same final exam and used the same textbook but different instructors assigned different homework problems and gave different hour exams, where hour exams were in-class, hour-long exams given approximately monthly. (Hour exams are often called "quizzes" or "midterms" at other schools.) In sections taught by Andes developers (Profs. Robert Shelby, Donald Treacy and Mary Wintersgill), students did their homework on Andes.

Each year, the Andes instructors recruited some of their colleagues' sections as Controls. Students in the Control sections did the same hour exams as students in the Andes section. Control sections did homework problems that were similar but not identical to the ones solved by Andes students.

The USNA requires all its 2nd-year students to take physics, so the students in the experiment could be considered a random sample of USNA sophomores. USNA students are academically well prepared (84% scored above 600 on the MSAT in 2010). They are highly motivated because all students receive 100% scholarships, and if they graduate, then they receive a commission in the US Navy or Marines. On the other hand, they often do have enough time to do all their coursework because they have many required non-academic activities as well.

Both Andes and Control instructors motivated their students to do their homework by scoring it on "effort displayed" rather than competence or correctness. Andes instructors had student submit their Andes solutions and used the Andes score as an "effort" score . On the other hand, the Control instructors initially marked the homework carefully in order to stress that the students should write proper derivations, including drawing coordinate systems, vectors, etc. Later in the semester, the Control instructors grade homework more lightly with short comments like "Draw FBD", "Axes" or "Fill out F=MA by components" in order to continue the emphasis on proper derivations. In some classes, instructors gave a weekly quiz (not an hour exam) consisting of one of the problems from the preceding homework assignment. This encouraged students to both do the assignments carefully and to study the solutions that the instructor handed out.

The same final exams were given to all students in all sections. The final exams comprised approximately 50 multiple choice problems to be solved in 3 hours. The hour exams had approximately 4 problems to be solved in 1 hour. Thus, the final exam questions tended to be less complex (3 or 4 minutes each) than the hour exam questions (15 minutes each). On the final exam, students just entered the answer, while on the hour exams, students showed all their work to derive an answer. On the hour exams, students were scored on (a) their vector drawing, (b) whether they defined variables and/or used standard, legible variable names and used them in preference to numbers in equations, (c) correctness of the equations written, (d) the correctness of the answer. Students were informed of the grading rubric well in advance.

The instructors believe that the open-response, complex problems of the hour exams were a more valid assessment of students' competence, but the multiple-choice final exam were necessary for practical reasons. The hour exam results will be reported first.

21.5.2 Hour Exams Results

It is important to check that the prior knowledge and skill of the Andes students was approximately the same as the prior knowledge and skill of the Control subjects. Students were not assigned randomly to condition, although it could be argued that they were assigned randomly to sections. Students were not given an appropriate pre-test. Thus, the standard methods for insuring equal prior competence were not used. However, over many years of experience, instructors have found that the students' physics grades are strongly correlated with their overall college grade point average (GPA) and their college major. Thus, one method of checking the equality of prior competence is to check that the two conditions have equal distributions GPAs and majors. A t-test showed no reliable differences in GPA in any of the years for which GPA data were available. In order to check for equivalent distribution of majors, the majors were classified as either Engineering, Science or Other. A 3x2 Chi-squared test confirmed that the two conditions had the equivalent distribution of majors in every year for which data were available. Thus, it seems likely that the Andes and Control group students had similar initial competence in physics.

Table 21.1 shows the hour exam results for all 5 years. It presents the mean score (out of 100) over all problems on one or more exams per year. In all years, the Andes students scored reliably higher than the Control students with moderately high effect sizes.[1] The 1999 evaluation had an effect size that was somewhat lower, probably because Andes had few physics problems and some bugs, thus discouraging students from using it. It should probably not be considered representative of Andes' effects, and will be excluded from other analyses in this section.

In order to calculate an overall effect size, it was necessary to normalize the scores across years so that they could be combined. The raw exam scores for each exam were converted to z-scores, and then the data points from years 2000 through 2003 were aggregated. The mean Andes scores was 0.22 (standard deviation: 0.95) and the mean score of Control students was –0.37 (0.96). The difference was reliable ($p < .0001$) and the effect size was 0.61.

[1] Effect size was defined as (AndesMean – ControlMean) / ControlStandardDeviation.

Table 21.1 Results from hour exams evaluations

Year	**1999**	**2000**	**2001**	**2002**	**2003**
Number of Andes students	173	140	129	93	93
Number of control students	162	135	44	53	44
Andes mean (stan. dev)	73.7 (13.0)	70.0 (13.6)	71.8 (14.3)	68.2 (13.4)	71.5 (14.2)
Control mean (stan. dev)	70.4 (15.6)	57.1 (19.0)	64.4 (13.1)	62.1 (13.7)	61.7 (16.3)
P(Andes = Control)	0.036	< .0001	.003	0.005	0.0005
Effect size	0.21	0.92	0.52	0.44	0.60

It is often the case that educational software helps some types of students more than other types. For instance, it is often found that students with high prior knowledge learn equally well from the experimental and control instruction, but students with low prior knowledge learn more from the experimental instruction than from the control instruction. In order to check for such aptitude-treatment interaction, we compared linear regression models of GPA vs. hour exam scores. We used the aggregated z-scores for the exams over years 2000 through 2002, because we lacked GPAs of student in the 2003 Control condition. Figure 21.5 shows the results. The regression lines of the two conditions are nearly parallel, indicating that there was little aptitude-treatment interaction. In a separate analysis, we found that the benefits due to Andes vs. Control did not depend on the student's major (VanLehn et al. 2005): Science, Engineering and Other majors all benefited equally.

The USNA physics instructors believed that the point of solving physics problems is not to get the right answers but to understand the reasoning involved, so they used a grading rubric for the hour exams that scored the students' work in addition to their answers. In particular, 4 subscores were defined (weights in the total score are shown in parentheses):

- *Drawings:* Did the student draw the appropriate vectors, coordinate systems and bodies? (30%)
- *Variable definitions:* Did the student use provide definitions for variables or use standard variable names? (20%)
- *Equations:* Did the student display major principle applications by writing their equations without algebraic substitutions and otherwise using symbolic equations correctly? (40%)
- *Answers:* Did the student calculate the correct numerical answer with proper units? (10%)

Andes was designed to increase student conceptual understanding, so we would expect it to have more impact on the more conceptual subscores, namely

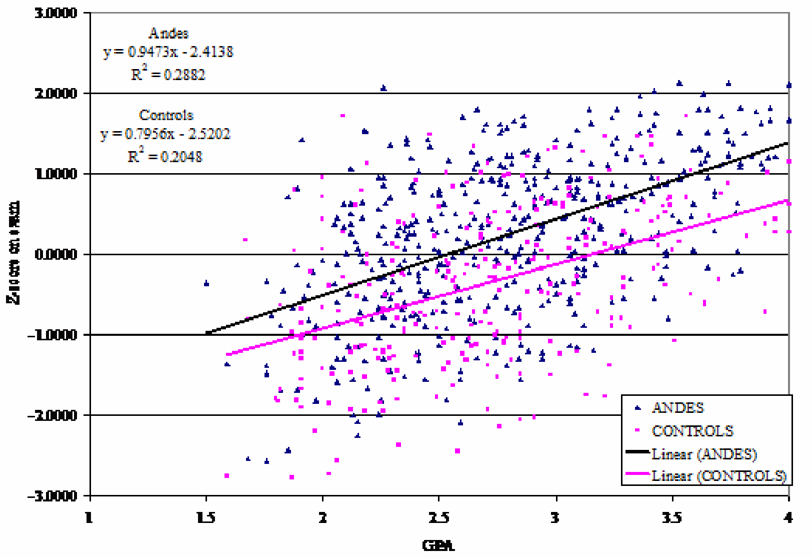


Fig. 21.5 Aptitude-treatment interaction: GPA (x-axis) vs. hour-exam score (y-axis)

Table 21.2 Hour exam subscore effect sizes (and p-values for t-tests)

	2000	2002a	2002b	2003	Average
Drawings	1.82 (<.001)	0.49 (.003)	0.83 (<.001)	1.72 (<.001)	1.21
Variable definitions	0.88 (<.001)	0.42 (.009)	0.36 (.026)	1.11 (<.001)	0.69
Equations	0.20 (.136)	0.12 (.475)	0.30 (.073)	-0.17 (.350)	0.11
Answers	-0.10 (.461)	-0.09 (.585)	0.06 (.727)	-0.20 (.154)	-0.08

the first 3. Table 21.2 shows the effect sizes, with p-values from two-tailed t-tests shown in parentheses. Results are not available for 2001. Two hour exams are available for 2002, so their results are shown separately.

There is a clear pattern: The skills that Andes addressed most directly were the ones on which the Andes students scored higher than the Control students. For two subscores, Drawing and Variable definitions, the Andes students scored significantly higher than the Control students in every year. These are the problem solving practices that Andes requires students to follow, because it requires quantities to be defined before they are used in equations. Although these requirements restrict the students' freedom, the instructors' insistence that Andes impose this restriction paid off in increased scores.

The third subscore, Equations, assesses a practice that is not required but it is encouraged by Andes' scoring policy. The effect sizes here were moderate and not statistically reliable. Perhaps Andes should have curtailed freedom even more and made proper usage of symbolic equations a requirement.

The Answers subscore was the same for both groups of students for all years even though the Andes students produced better drawings and variable definitions on those tests. This suggests that the probability of getting a correct answer depends strongly on other skills, such as algebraic manipulation, that are not measured by the more conceptual subscores and not emphasized by Andes. On the other hand, the tied Answer subscores suggest that the Andes students' use of the equation solving tool did not seem to hurt their algebraic manipulation on the hour exams.

In summary of the hour exam results, Andes students tended to learn more than Control students, with an overall effect size of 0.61. Andes was equally beneficial for high-GPA and low-GPA students and for different majors. Breaking down the hour exam scores into subscores showed that Andes students scored higher than Control students on drawing and variable usage, which are two fundamental conceptual skills emphasized by Andes. Andes does not teach algebraic manipulation skills and only encourages proper use of symbolic equations, so it had little apparent advantage over paper for teaching these skills.

21.5.3 Final Exam Scores

Although the hour exams used complex, open-response problems instead of multiple choice problems, and this allowed instructors to use a grading rubric that valued conceptually clear derivations, the final exams were scored on the basis of the student's final answers, and most of the final answers were multiple choices, as are many high stakes assessments. Although the answer-only format probably reduces the validity of the final exams, students studied very hard for them, and we were interested in whether the benefits of Andes would still be visible after such intensive studying.

The final exams covered the whole course, but Andes did not. However, Andes' coverage steadily increased over the years. In 2003, Andes covered 70% of the homework problems in the course. This section reports an analysis of the 2003 final exam data.

First, we need to check that the Andes sections have a representative sample of the whole course population. As mentioned earlier, GPA and major seem to be the best measures of prior knowledge and skill. The Andes students' mean GPA was 2.92 (SD = 0.58), which was marginally significantly higher ($p = 0.0662$) than the non-Andes students' mean GPA of 2.80 (SD = 0.55). Moreover, the distribution of majors among the Andes students was statistically different from the distribution of majors among the non-Andes students ($p < .0001$, 3x2 Chi-squared test, where majors were aggregated into 3 types: engineering, science and other). In particular, there were relatively more engineers than in the Andes sections than in the non-Andes sections. Thus, it appears that the Andes students were *not* representative of the whole population. Thus, we had to use statistical techniques to factor out effects of the higher prior knowledge of the Andes students.

For each group of majors, we regressed the final exam scores of all students in the course against the students' GPAs. Of the 931 students, we discarded scores from 19 students with unclassifiable majors or extremely low scores. This yielded three statistically reliable 2-parameter linear models, one for each type of major. Each model expresses the relationship between general competence and the final exam score. For each student, we subtracted the exam score predicted by the linear model from the student's actual score. This residual score represents how much better or worse this student scored compared to the score predicted solely on the basis of their GPA and their major. That is, the residual score factors out the students' general competence. The logic is the same as that used with an ANCOVA, with GPA and major serving as covariates instead of pre-test scores.

Using these residual scores, we evaluated Andes' impact on students in each of the 3 groups of majors. As Table 21.3 indicates, the residual scores of the engineering and science majors were not statistically different with Andes than with paper homework. However, the other majors did learn more with Andes than with paper homework ($p<.016$; effect size = 0.5). Over all students, the Andes students mean residual score was higher than the mean residual score of the non-Andes students (effect size = 0.25; $p=0.028$).

Although the overall effect of Andes was positive and reliable, Andes had smaller benefits on the learning of the engineering and science majors for two plausible reasons. (1) The engineering majors were concurrently taking a course on Statics, which has very similar content to the physics courses. This dilutes the effect of Andes, since it affected only their physics homework and not their Statics homework. (2) Both the science and engineering majors probably took advanced physics courses in high school, and thus may have developed considerable skill at obtaining correct answers without showing conceptually explicit derivations. This would explain why the Andes engineering and science majors did better than the controls on the hour exams, which were scored conceptually, but they did not do better than the non-Andes students on the final exam, which was not scored conceptually.

Thus, Andes students overall learned significantly more than non-Andes students. The overall effect size was somewhat smaller for the final exam (0.25) than the hour exams (0.61). This may be partially due to the fact that roughly 30% of the final exam addressed material not covered by Andes. It may also be partially due to the format of the final exam. The final exam had students enter only their answers, whereas the hour exams had students show their work, which allowed graders to assess their conceptual understanding more directly.

Although the overall effect of Andes was positive and reliable, Andes had smaller benefits on the learning of the engineering and science majors for two plausible reasons. (1) The engineering majors were concurrently taking a course on Statics, which has very similar content to the physics courses. This dilutes the effect of Andes, since Andes affected only their physics homework and not their Statics homework. (2) Both the science and engineering majors probably took advanced physics courses in high school, and thus may have developed considerable

Table 21.3 Residual scores on the 2003 final exam

	Engineers	**Scientists**	**Others**	**All**
Number of Andes students	55	9	25	89
Number of non-Andes students	278	142	403	823
Andes students mean (stand. dev.)	0.74 (5.51)	1.03 (3.12)	2.91 (6.41)	1.38 (5.65)
Non-Andes students mean (s.d.)	0.00 (5.39)	0.00 (5.79)	0.00 (5.64)	0.00 (5.58)
p(Andes=non-Andes)	0.357	0.621	0.013	0.028
Effect size	0.223	0.177	0.520	0.25

skill at obtaining correct answers without showing conceptually explicit derivations. This would explain why the Andes engineering and science majors did better than the controls on the hour exams, which were scored conceptually, but they did not do better than the non-Andes students on the final exam, which was not scored conceptually.

Thus, Andes students overall learned significantly more than non-Andes students. The overall effect size was somewhat smaller for the final exam (0.25) than the hour exams (0.61). This may be partially due to the fact that roughly 30% of the final exam addressed material not covered by Andes. It may also be partially due to the format of the final exam. The final exam had students enter only their answers, whereas the hour exams had students show their work, which allowed graders to assess their conceptual understanding more directly. There could be other explanations as well.

21.5.4 Discussion of the Evaluations

This section summarizes the evaluation results and compares them to those from the Koedinger et al. (1997) study that tested a combination of an intelligent tutoring system (PAT) and a novel curriculum (PUMP). A revised version of this combination is now distributed by Carnegie Learning as the Algebra I Cognitive Tutor (www.carnegielearning.com). There are only a few in-school, semester-long, controlled studies of intelligent tutoring systems in the peer-reviewed literature, and the widely-cited Koedinger et al. (1997) study is arguably the benchmark study against which others should be compared.

Using experimenter designed tests of conceptual understanding, Koedinger et al. (1997) found effect sizes of 1.2 and 0.7. The Andes hour exams were intended to measure both conceptual understanding and algebraic manipulation skills. When the hour exam scores were broken down into conceptual and algebraic components, Andes students scored significantly higher on the conceptual components that Andes addressed (Diagrams: effect size 1.21; Variables: effect size 0.69). In this respect, the results from the two studies are remarkably similar.

Standard tests may be less sensitive to conceptual understanding due to their final-answer format and different coverage. If so, one would expect them to be less sensitive to the benefits of tutoring. Indeed, the Koedinger et al., (1997) evaluation found smaller effects when using multiple-choice standardized tests: 0.3 for each of two tests. These results are comparable to our results for the multiple-choice final exam, where Andes students scored higher than non-Andes students with an effect size of 0.25. The 0.05 difference in effect size may be due to the facts that 30% of the homework problems are not yet covered by Andes.

Thus, the Andes evaluations and the Koedinger et al. (1997) evaluations have remarkably similar effect sizes: 1.2 and 0.7 on experimenter-designed tests and 0.3 on standard tests.

The Andes evaluations differed from the Koedinger et al. (1997) evaluation in a crucial way. The Andes evaluations manipulated only the way that students did their homework—on Andes vs. on paper. The evaluation of the Pittsburgh Algebra Tutor (PAT) was also an evaluation of a new curriculum, developed by the Pittsburgh Urban Mathematics Project (PUMP), which focused on analysis of real world situations. It is not clear how much gain was due to the tutoring system and how much was due to the reform of the curriculum. In our evaluation, the curriculum was not reformed. Indeed, the Andes students and the Control students were in the same course and used the same textbook. The gains in our evaluation are a better measure of the power of intelligent tutoring systems per se. This is extremely good news.

21.6 Progress toward Scaling up Andes

This chapter has made three of its four points so far. First, it discussed how Andes allows more freedom in problem solving than most ITS. Second, it discussed the challenges of scaffolding students in such a freeform user interface and presented Andes' solutions. Third, it discussed the evaluation results, which showed that Andes was unequivocally beneficial in a long-term, controlled evaluation of learning.

The last topic is to report the progress on scaling up Andes. This has proved to be the most difficult goal of all. Briefly put, we have been "selling" Andes for many years by presenting talks and booths at Physics Education Conferences, networking with instructors, and making countless personal contacts and visits. At this writing, Andes2 has a small set of instructors who use it every semester in their classes. However, tens of instructors have tried it in their classes for one semester and dropped it, and hundreds of instructors have worked Andes problems for several hours then decided not to adopt it in their classes. In many causes, we have interviewed the non-adopters and elicited their views on Andes' strengths and weaknesses.

First the good news. The non-adopters approved of Andes main design goal, which was to provide students with as much freedom as paper and pencil. Even the few restrictions that Andes imposes on freedom (e.g., immediate flag feedback; defining quantities before using them in equations) met with general approval. Moreover, non-adopters were mostly satisfied with the approximately 500 Andes physics problems available even though textbooks have about 2000 problems. Our ability to rapidly add new problems to the system was only occasionally used.

Non-adopter's major reason given for abandoning Andes was that Andes2 could only be run on Windows and it had to be installed. This meant that Mac users had to use an emulator, and that users of public machines (e.g., in school's computer labs) had to either reinstall Andes at each use or have tech support preinstall it on the public machines.

The second major reason given for abandoning Andes was that it was not commercially supported software. Potential adopter were worried that user support might disappear. Nothing we could say would dislodge the concept that software companies are more stable than research groups, even a research group supported by large, long-term NSF center (the Pittsburgh Center for the Learning Sciences).

Lastly, the Andes server logs indicate a large number of casual users who try Andes but give up soon. We have spoken to only a few of these non-adopters. Their major barrier was the complex user interface of Andes2.

Given this clear feedback from non-adopters, Andes3 was developed and is about to be launched.

To allay the first concern (Windows only; installation necessary), Andes3 is a client-server system where the client is written in Javascipt using Ajax techniques. That is, any student whose machine has a browser that is connected to the internet can run Andes by clicking on a URL that points to an Andes problem page. The page downloads like any other HTML resource (i.e., it passes through most machine and cluster firewalls). The page displays an unsolved Andes problem, and waits for the user to click on one of Andes' buttons. When the student makes an entry (e.g., by clicking on a text-entry tool, typing, and pressing the Enter key), the raw entry is sent to the Andes server. The server sends back a small change to the page being displayed so that an entire new page doesn't need to downloaded. This is the same technology used by Google Maps and other recently developed client-server software. The Andes3 client has been tested on a large number of combinations of browsers and platforms. The Andes3 server has been load tested with up to 300 (simulated) users concurrently. Andes3 appears ready for prime time.

To allay the second concern (no apparent long-term support), Andes3 has been configured to run under several physics learning management systems that have provided trusted service for years. We are currently working with WebAssign (www.webassign.com) and Lon-Capa (www.lon-capa.org) as well as our long-time collaborators at the Open Learning Initiative (www.cmu.edu/oli). With its new architecture, Andes problems can be mixed in with physics problems supported by more conventional software (e.g., students enter a final answer and submit it to server for grading). Because these organizations have reputations for providing good service to their users, Andes3 is undergoing testing by them. Moreover, all its code has been made open source, so that these organizations can take over support if anything should happen to us.

To allay the third concern (a complex user interface to learn), the Andes3 user interface was designed to mimic the Microsoft PowerPoint user interface, which many physics instructors and students are already familiar with. This was trickier than it seems, because similarity needs to extend down to the details of selecting, dragging, resizing, deleting, double clicking, etc. For instance, what should happen when a user clicks on the arrow-drawing tool, then simply clicks on the canvas instead of dragging out an arrow? It turns out that PowerPoint draws a down-ward pointing arrow about an inch long. So Andes3 should too, as there are likely to be some users who would expect that.

In short, even though Andes2 has been used successfully in many year-long physics courses by instructors who did not participate in its development, and evaluations have shown that it raises students exam grades by 0.61 standard deviations, a few supposedly minor issues appear to have thwarted its dissemination. It will be interesting to see if removing them allows Andes to become widely used.

Acknowledgements. This research was supported by the Cognitive Sciences Program of the Office of Naval Research under grants N00019-03-1-0017 and ONR N00014-96-1-0260, and by the NSF under grant 0354420 to the Pittsburgh Science of Learning Center. We gratefully acknowledge the Andes Alumni: Patricia Albacete, Cristina Conati, Ellen Dugan, Abigail Gertner, Collin Lynch, Zhendong Niu, Charles Murray, Joel Shapiro, Kay Schulze, Stephanie Siler, Linwood Taylor, Donald Treacy, Anders Weinstein and Mary Wintersgill. Some text, tables and figures were copied from the *International Journal of Artificial Intelligence and Education*, 15(3), K. VanLehn et al., “The Andes physics tutoring system: Lessons learned,” pp. 147-204, copyright 2005, with permission from IOS Press.

References

Albacete, P.L., VanLehn, K.: The Conceptual Helper: An intelligent tutoring system for teaching fundamental physics concepts. In: Gauthier, G., VanLehn, K., Frasson, C. (eds.) ITS 2000. LNCS, vol. 1839, p. 564. Springer, Heidelberg (2000a)

Albacete, P.L., VanLehn, K.: Evaluation the effectiveness of a cognitive tutor for fundamental physics concepts. In: Gleitman, L.R., Joshi, A.K. (eds.) Proceedings of the Twenty-Second Annual Conference of the Cognitive Science Society. Erlbaum, Mahwah (2000b)

Aleven, V., Koedinger, K.R.: Limitations of student control: Do students know when they need help? In: Gauthier, G., VanLehn, K., Frasson, C. (eds.) ITS 2000. LNCS, vol. 1839, p. 292. Springer, Heidelberg (2000)

Anderson, J.R., Corbett, A.T., Koedinger, K., Pelletier, R.: Cognitive Tutors: Lessons Learned. The Journal of the Learning Sciences 4(2), 167–207 (1995)

Baker, R.S., Corbett, A., Koedinger, K.R., Evenson, S., Roll, I., Wagner, A.Z., et al.: Adapting to when students game an intelligent tutoring system. In: Frasson, C., Gauthier, G., McCalla, G.I. (eds.) ITS 2006. LNCS, vol. 4053, pp. 392–401. Springer, Heidelberg (2006)

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

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

Brna, P., Caiger, A.: The application of cognitive diagnosis to the quantitative analysis of simple electrical circuits. In: Frasson, C., Gauthier, G., McCalla, G.I. (eds.) ITS 1992. LNCS, vol. 608. Springer, Heidelberg (1992)

Gertner, A., Conati, C., VanLehn, K.: Procedural help in Andes: Generating hints using a Bayesian network student model. In: Proceedings of the 15th national Conference on Artificial Intelligence (1998)

Gertner, A.S., VanLehn, K.: Andes: A coached problem solving environment for physics. In: Gautheier, G., Frasson, C., VanLehn, K. (eds.) ITS 2000. LNCS, vol. 1839, p. 133. Springer, Heidelberg (2000)

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

Kane, D., Sherwood, B.: A computer-based course in classical mechanics. Computers and Education 4, 15–36 (1980)

Koedinger, K.R., Anderson, J.R., Hadley, W.H., Mark, M.A.: Intelligent tutoring goes to school in the big city. International Journal of Artificial Intelligence in Education 8(1), 30–43 (1997)

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

Mitrovic, A.: An intelligent SQL Tutor on the Web. International Journal of Artificial Intelligence and Education 13(2-4), 171–195 (2003)

Mitrovic, A., Ohlsson, S.: Evaluation of a constraint-based tutor for a database language. International Journal of Artificial Intelligence and Education 10, 238–256 (1999)

Nicaud, J.F., Bouhineau, D., Varlet, C., Nguyen-Xuan, A.: Towards a product for teaching formal algebra. In: Lajoie, S.P., Vivet, M. (eds.) Artificial Intelligence in Education. IOS Press, Amsterdam (1999)

Norman, D.A.: Categorization of action slips. Psychological Review 88(1), 1–15 (1981)

Rose, C.P., Roque, A., Bhembe, D., VanLehn, K.: A hybrid language understanding approach for robust selection of tutoring goals. In: Cerri, S.A., Gouardéres, G., Paraguaçu, F. (eds.) ITS 2002. LNCS, vol. 2363, pp. 552–561. Springer, Heidelberg (2002)

Shapiro, J.A.: Algebra subsystem for an intelligent tutoring system. International Journal of Artificial Intelligence in Education 15(3), 205–228 (2005)

Shute, V.J.: A macroadaptive approach to tutoring. Journal of Artificial Intelligence in Education 4(1), 61–93 (1993)

Singley, M.K.: The reification of goal structures in a calculus tutor: Effects on problem solving performance. Interactive Learning Environments 1, 102–123 (1990)

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

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

Yibin, M., Jinxiang, L.: Intelligent tutoring system for symbolic calculation. In: Frasson, C., McCalla, G.I., Gauthier, G. (eds.) ITS 1992. LNCS, vol. 608. Springer, Heidelberg (1992)