A Computer Tutorial System for Introductory Physics Courses

Joel A. Shapiro

Abstract

Problem solving by students is an essential part of any introductory course in analytical physics. Unfortunately, large numbers of would-be engineers, bio-sci majors and others who need such analytic courses fail to benefit from homework problems because they do not get the help they need in attacking them. As a consequence, we tend to assign only quite simple problems the students can get through without assistance, but these reinforce the "search for the formula to plug into" approach, rather than encouraging analysis. Computer guided problem solving has great potential to substitute for the missing human tutoring, enabling students to attack more demanding material.

This talk demonstrates the ANDES tutorial system, on which I worked last year, which is in use at the US Naval Academy in their introductory course, which all students must take.

Introduction

1 Title

Today I am going to tell you about an Intelligent Tutoring System (ITS) for helping students develop problem-solving skills in a quantitative introductory physics course. Such a system is not just a homework grader — (2 Intelligent Tutoring Systems) it does more than mark a student's answer right or wrong. Ideally, an Intelligent Tutoring System needs to model the understanding a student has of an involved domain, figuring out from the answers a student gives what parts of the problem she understands, and what misconceptions she is displaying. The tutor can give guidance, indicating correct vs. incorrect steps, providing "what's wrong with that?" help, and also giving help when a student is stuck and doesn't know how to proceed.

Most of us believe that learning physics without solving physics problems cannot lead to a very satisfactory understanding. 3 Learning by Solving Problems But it is also nearly universally acknowledged that a student learns very little by watching someone else do a physics problem — the student needs to be trying to do the problem herself to have meaningful involvement and in order for meaningful learning to take place. Thus all quantitative physics courses assign homework problems, which are intended to provide this exposure. What we hope the students will learn is a process of analysing a situation, decomposing a problem into semi-independent pieces, examining which fundamental principles apply, and utilizing these, usually in the form of equations, to set up the basic pieces of a solution, and then construct from those pieces the full solution.

4 What not to learn

Unfortunately, doing the kind of analysis just mentioned is hard for our students. Many students enter our courses with coping skills that depend heavily on a technique which searches for a formula to plug the given values into. A problem requiring decomposition strains their abilities, and soon the student finds herself unable to make progress. One adaptation to this situation is to resign ourselves to asking only or primarily questions for which this formula-search method will work. Unfortunately this very much limits the kind of learning we can achieve. It also encourages students to stick to their weak methods rather than developing the stronger intellectual methods needed to really understand physics. What is needed for deeper learning is interaction with a tutor who can guide the student to make progress on a complex problem that requires analysis. This is something most human tutors are able to do effectively, but this is quite difficult to build into a computer program. Quite difficult, but not impossible. I am going to describe a project on building such a system, called ANDES, 5 Andes II which is already partially successful and has great promise of being a robust and effective tutor.

6 Andes Developers

The Andes system was begun about five years ago in a joint project of Kurt VanLehn, of the Learning Research and Development Center of the University of Pittsburgh, and the Physics and Computer Science Departments of the US Naval Academy. All students of the Naval Academy have to take the same introductory physics course, which is taught (last year from Serway and Beichner) at roughly the level of our 123 or 203 courses. 7 Andes Although the program is still rather provisional and rapidly evolving, it has been used for several years in this course; six sections in the fall of 2000 did their homework in Andes.

The current version of Andes supports problems in introductory mechanics in two dimensions, roughly through Chapter 12.3 in Serway and Beichner. Here is the list of topics 8 Andes topics. It is expected that the coverage will grow with time, with circuits the next priority. From the Andes screen shown, one chooses a topic. I will explain what Andes does by an example, a common problem which is a combination of a mass on an incline with Atwood's machine[1]. This topic is part of **translational dynamics**, so after selecting that topic 9 Select problem we get a list of all the problems in that topic. Here is a blowup 10 Problem choosing blowup so you can see it. Note that the status of the student's solutions of them, and a view of whichever one is being pointed to. Clicking on one problem brings us to the opening screen for that problem 11 Opening problem presentation There is a verbal problem description, a diagram, and areas devoted to variable definitions (upper right), equation entry (lower right), drawing (on the diagram) and dialog (lower left). We will see how these work as we go along.

12 An example problem Here is a version of the problem statement that you can actually read. `[PAUSE]`

13 Problem decomposition

Working this problem requires a decomposition into quite a number of parts:

- Draw the free body diagram for the mass on the slope
- Apply Newton's second law along the slope.
- Draw the free body diagram for the hanging mass

[1]For example, Serway and Beichner Example 5.10

- Apply Newton's second law to the hanging mass vertically.
- recognize that the Tensions are the same, and that the accelerations are corrolated
- Use all of the above to find the acceleration
- Use the kinematics of constant acceleration to find the distance the hanging mass falls in two seconds.

Without help, many students are likely to make a mistake somewhere along the way, and looking only at the final numerical answer is not likely to provide adequate clues to where the student has gone wrong. Andes encourages, and perhaps requires, that the student declare variables and write down equations for solving the problem. In our example, [14 Selecting a body] we will need to make statements that might be time dependent. Time points are predefined in Andes, here `T0` and `T1` are the initial and final times. A [15 Selecting body (blowups)] good first step is to begin by considering the block on the incline, and draw a free-body diagram for it. In Andes, one selects a body with the "body tool", the dot in the tool bar on the left side. Andes asks us which body we are dealing with, and over what time period what we define is valid.

Next we begin defining the forces [16 Defining a force, part 1] on the selected body with the force tool $\vec{F}$ from the left hand toolbar, and draw it with the mouse. As soon as it is drawn, dialog boxes insist the student specify completely what force she is defining, [17 Specifying force's agent] not only the body acted on, but what the agent is, and the type of force, [18 Specify type]. There is a box which allows making fine corrections to the direction of the force — Note that the force I drew was at 27°, even though I wanted 25°. I can correct it here, and also change the name of the variable. Finally, I need to give [19 Finishing up force] the time interval over which it acts. Note the changed angle and the variable name used for the magnitude of the force. The dialog box suggests at this point that the student define axes, (although in the slide this is partly obscured). That is done [20 Defining axes (blowup)] using the "axes tool", second button on left hand toolbar. The direction is set with the mouse, and corrected if necessary in the dialog box (not shown). 21 Defining axes When axes have been selected, each vector item defined automatically develops variables for its x and y components.

22 Mistaken force

We continue by drawing the weight force, and then we make a common error, thinking that normal forces always point up, rather than perpendicular to the surface. 23 Normal straight up? Up to now, all our entries have turned green as soon as submitted, but wrong ones turn red. This not only provides immediate feedback, but enables the student to ask for help. 24 Asking what's wrong Right-clicking on the wrong vector brings up a menu. Let's get a closer look: 25 What's wrong (blowup) We will select "What's wrong with that?". 26 Hints on normal forces The system first gives a vague suggestion, 27 Hints (blowup) which the student could repair without further dialog if she sees the error of her ways, but if not, "explain further" gives a clear explanation of what is wrong and how to fix it. As currently configured, hints continue until one is given that even an idiot could follow, so as to proceed to the next step. I will spare you that, and correct the normal force 28 Defining an acceleration and now I will draw the acceleration of block 1. For defining accelerations, 29 Acceleration dialog we need to specify the object, whether we are talking about an instanteous or average acceleration, at which time or over which time interval, and, of course, define a variable for its magnitude, correcting the direction if necessary.

30 Entering equations

At this point we should be ready to start writing down equations. There is a lot more variation in acceptible equations than you might imagine. 31 Eqs. for block on incline For example, the USNA professors would have preferred the first equation to have been

$$Fta_x + Fw_x(+Fn_x) = ma \times aa_x,$$

to be followed by equations which give these x components in terms of the magnitudes and directions. But while Andes developers do seem to be quite devoted to such "tight scaffolding" for guiding the student, Andes does not require it. In fact, various levels of scaffolding may be required, and it is intended that this scaffolding will "fade" as the student develops competance.

Notice that the student must enter the required given values, including the one given implicitly, which in the specification of this problem to the computer system declares that all this is taking place near the surface of the Earth. The student needs to specify $g = 9.8\text{m/s}^2$, without forgetting the units, without which her equation will be marked wrong.

Having completed the analysis of the block on the plane, 32 The second body it is time to focus on the hanging block. This requires the same steps as for the block on the incline, so I did them without comment.

33 Solving for the acceleration

Next we enter the two equations stating that the tensions are the same and the magnitudes of the accelerations are the same. We now have enough information to solve for the acceleration. The USNA academy instructors feel that their job is simply to teach the physics principles, not give the students drill in algebra, so they are very pleased that there is a "solve-tool" built in. What you see is the student asking the "solve-tool" to solve for the acceleration `ab`. Here is what the whole screen looks like just before we let go:

34 Andes solves equations

I had many discussions with the Naval Academy folk on whether this was really a good idea. We agreed that it was for humanity majors but probably not for engineers, and we should provide a more limited tool for the engineers. This is not fully implemented, because the USNA people are not very interested, but I can discuss what I think it should do if there is interest.

Anyway, when we let go Andes will solve for `ab` and write the equation in the first unoccupied slot, (which makes my idea of separating the parts with blank equations not seem so smart) so look at line 5 for the answer. Here, at the top, is a version you can actually see. 35 Undefined variable Andes doesn't have its significant figure analysis very well developed, so it just gives all the accuracy it has.

After getting the solution for `ab`, the student now turns to the distance dropped, writing the equations $d = \frac{1}{2}ab \times t^2$, but Andes immediately objects to the use of the undefined variable d.

This was just an oversight, so the student defines it using the displacement vector tool, specifying information similar to that for acceleration, and then reenters the equation, which turns green.

36 Dialog

Then the solve-tool is asked for d, and writes the answer in the next available slot, number 9. The student recognizes this is the answer, so, truncating to a reasonable accuracy, places the answer in the answer box, 37 Answers need correct units but it turns red, and the dialog box complains that the student forgot to put units on the answer. Once the student corrects this, the answer turns green. 38 Right answer At this point the student is done with the problem. She can print out the page, which produces a version with less cropping. `[PASS AROUND]` At the USNA, the students turn this in as their homework. The student then closes the problem, 39 Problem closed which brings back the "Select problem within topic" page, with the problem just completed shown as done, and the cursor set to open the next uncompleted problem. If the student is done working with Andes at the moment, she exits. But before the program exits, 40 Send log of session it first pops up a window asking for a log of the session to be sent. This is currently used only by the developers, not as part of the grading or teaching of the student. I think it might be used for grading, though, if the grading software were written.

I have now worked through a typical, perhaps somewhat harder than average, problem in Andes with you. At this point, there are two more things we might do. 41 To do next?

- Describe what it takes to make problems in Andes. This will give you a feeling for how different Andes is from a program like WebAssign.
- Switch to Andes itself, and have one of you, or the group as a whole, try to solve another Andes problem.

Entering new problems in Andes

Andes is a system in development and by no means a finished project. But it is designed in a way which should allow use over a very broad range of problems, which might be specified by people other than those responsible for building in the physics knowledge. [42 Andes structure] The components of Andes are complex. It has a physics knowledge database which contains such concepts as Newton's Laws and Momentum Conservation, and actually uses these to solve the problems which are defined for it. This is totally unlike a system like WebAssign, which understands nothing of the physics, but only how to display the question and read the answers, with some instructions from the instructor about the range of acceptible answers.

Thus a problem is specified to Andes as a problem statement, not a solution. The statement needs to be more fully and unambiguously specified than in the homework problems in a book, but nonetheless it is only a stricter and more formal specification than the one in the book. Andes then solves the problem, often in many different possible ways.

Just to give an idea of how the system works, I will show you the problem specification for the problem we just did, known to Andes as **exdt2a**. Unfortunately this stuff can be read only by people who love `lisp`, but I will try to help you skim it.

[43 Prob. desc. (comments)]

The problem description begins with a defproblem, which has several fields. On this slide are only comments - the first is the problem statement that will appear to the student (but is actually done elsewhere for real). The :features is just help for the developers to keep track of the status and topic, and the comments shown here are rather technical, and in fact are also completely out of date. The real meat of the problem description is on the next slide.

[44 Prob desc (cont)]

The first line specifies what the student needs to supply as the answer. In this case, it is the magnitude of the displacement of the hanging block from time 1 to time 2 [You may notice inconsistencies which have developed over years of development with inadequate time to clean things up.]

The rest is the given information. First we specify the relevant time points in this problem, an initial time (1), a final time (2), and the time interval between them, [called (during 1 2), sorry about that - you've got to be lispy to love that].

The next line specifies that the time interval is 2 seconds. Then that there is an object called block 1, it has a mass of 30 kg, it is supported by a plane at an incline of 25° during the time interval [1,2], that it is tied to a string which goes off at an angle of 25°, and that its motion is constrained to that direction, all of this information holding for the entire time interval [1,2].

Next, after an apologetic comment because this shouldn't be needed, Andes is told that the acceleration of block 1 is constant for $t \in [1, 2]$. Then that this occurs near the Earth, which Andes knows means that there is a gravitational force and $g = 9.8\text{m/s}^2$.

Then we turn to object 2, giving the same kind of information, plus the statement that at time 1 it is momentarily at rest. It is understood that the strings referred to are the same, because they have the same name ("string").

Notice that the full specification is only 34 lines, of which 13 are comments, so specifying a problem a finite task, although clearly the language could be greatly improved by using a form more familiar to physicists.

Naturally new problems covering situations different from what has been dealt with before can easily involve requiring additions to the physics knowledge database. This is a task only for a real programmer, so right now there is no real push to make the problem specification more accessible. But in principle this will come.

Let's try Andes

I am now going to close PowerPoint and open Andes for real.