

## **2006-1199: FACILITATING THE DESIGN OF DIAGNOSTIC LEARNING MODULES WITH CAPE**

### **Larry Howard, Vanderbilt University**

Larry P. Howard is a Senior Research Scientist with the Institute for Software Integrated Systems at Vanderbilt University. He is the developer of the Courseware Authoring and Packaging Environment (CAPE) and the VaNTH experimental Learning Management System (eLMS).

### **Robert Roselli, Purdue University**

Robert J. Roselli is a Professor of Biomedical Engineering and Chemical Engineering at Vanderbilt University. He has served as Director of Graduate Studies for the Department of Biomedical Engineering, Education Director for the VaNTH Engineering Research Center in Bioengineering Learning Technologies, VaNTH Domain Leader in Biotransport, and is an active contributor to the VaNTH Biomechanics Domain. Dr. Roselli has developed graduate and undergraduate courses in biomechanics and biotransport at Vanderbilt University. He received B.S. (1969) and M.S. (1972) degrees in Mechanical Engineering and a Ph.D. (1975) in Bioengineering from the University of California, Berkeley.

# Facilitating the Design of Diagnostic Learning Modules with CAPE

## Abstract

Diagnostic learning modules adapt themselves in response to common difficulties experienced by students attempting problem-solving tasks. The nature of these adaptations concerns providing focused remediation and adjusting the flow of follow-on activities. An authoring environment that supports the design of such modules also provides facilities for capturing recurring structural and procedural idioms as *instructional design patterns*, with the aim of reducing the number of unique decisions to be addressed when creating similar modules. We report an application of this strategy that involves creating a general design pattern for numerical problem-solving tasks. New modules can be derived from this pattern by specifying details of an instance as data structures, thereby limiting what needs to be known about the design environment and its representation by novice designers. Our motivation is to expand the creation and use by faculty of these scaffolding designs by reducing the effort required to effectively employ the supporting technologies.

## Introduction

The Vanderbilt-Northwestern-Texas-Harvard-MIT Engineering Research Center for Bioengineering Educational Technologies (VaNTH ERC) has developed a learning technology infrastructure<sup>1</sup> that enables educators to create innovative online assignments which provide students with real time formative feedback as they attempt to solve quantitative engineering problems<sup>2</sup>. This infrastructure consists of a visual language-based authoring technology called CAPE (Courseware Authoring and Packaging Environment) and a web-based delivery platform called eLMS (experimental Learning Management System). Based on principles provided by the "How People Learn" framework, CAPE-designed modules have proven to be effective in improving learning in multiple areas including biotechnology, biomechanics and systems physiology<sup>2,3,4,5</sup>. The interactive system has found favor with instructors, teaching assistants and students. Students are allowed to attempt problems multiple times without being heavily penalized for making common mistakes, which can be diagnosed by the modules. Adaptive feedback provided by the modules helps students see what they may have forgotten to include in their solution. Students also like the automatic grading provided by such modules. Since each step taken by the student in the problem solution is instrumented by the eLMS delivery platform, students and instructors can easily review the module to support reflection. This approach also frees teaching assistants from the necessity of grading homework, most of which is worked correctly, and allows them to spend time with the students who most need their help.

Many educators recognize the value of such modules, but designing, constructing and debugging diagnostic learning modules using CAPE can be very time consuming. Educators will readily use modules designed by someone else, but claim to be too strapped for time to design their own diagnostic modules. The design environment that arises from this infrastructure provides a rich set of features and capabilities, but until recently, offered little in the way of scaffolds for designers. One way to attract new or less-experienced designers is to increase their efficiency by offering more structure in the design task. Integrative design environments can promote

consistencies among designs that contribute to design sharing among educators and improve usability by learners.

Our objective was to build a single CAPE design pattern that provides a consistent data-driven strategy for designing engineering problems. The same pattern can be reused for different problems, permitting instructors and teaching assistants with only a basic knowledge of CAPE to build diagnostic courseware. The author only needs to know how to sequence the input questions, construct the solution, and evaluate student responses. The pattern automatically cycles through any number of specified problems or questions. Input parameters can be randomized so each student works a unique numerical problem. The engine compares the student response to a correct response, or set of responses, and provides appropriate feedback to the learner for any number of diagnosed erroneous responses. Subsequent questions offered to the learner are based on the response to the previous question. The time and date that the assignment is due is enforced by the eLMS. Upon completion of the assignment, the learner is immediately provided with a grade and suggestions for seeking additional help, if needed. The learner can review their completed problem after it is submitted.

## Methods

Operation of the design pattern is quite straightforward and involves repetitive application of four fundamental steps: 1) pose a problem; 2) get student response; 3) evaluate response; and 4) provide diagnostic feedback. If the student response is correct the engine moves to the next question. If the response is incorrect, the student can be asked a new question or series of questions that are designed to discover where the student made an error. Alternatively, the author might elect to ask the same question for some maximum number of iterations, providing additional diagnostic information after each iteration. If the maximum number of attempts is exceeded, the author can elect to display the correct result(s) before moving on to the next question. After the final question has been answered, the student's grade will be displayed to the student if the author so chooses.

The CAPE model which accomplishes this is shown in Figure 1. The author need be concerned with providing data for only three structured collections of data, or *condition sets*. Condition Set (CS) 'Input' contains input parameters, problem statements, and the name of the first question to be presented to the student. Calculations that lead to the correct answer(s) and diagnosed incorrect answers are placed in CS 'Compute.' All of the information associated with each question and its associated diagnostics are placed in CS 'Assess.' The other condition sets in Figure 1 control model delivery and capture model output. Upon entry, the model is initialized by getting the name of the first

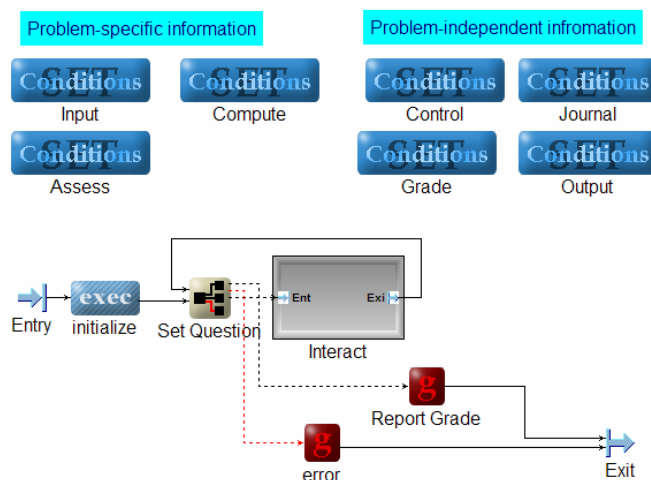


Figure 1. CAPE Design Pattern for reusable modules

assessment from 'Input.' If this is a valid assessment name, control is passed to the 'Interact' block. If it is not a valid assessment name, the module will provide an error message and stop. If there are no additional questions, the engine will report student grades and exit.

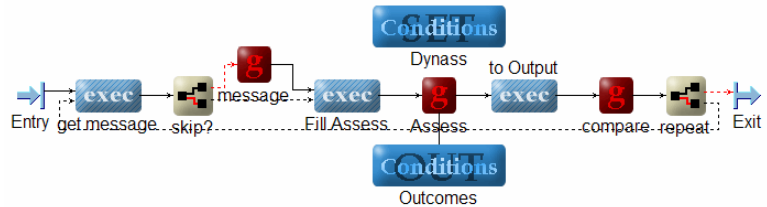


Figure 2. Details of the 'Interact' Block in Figure 1.

Details of the 'Interact' block are shown in Figure 2. A pre-assessment message providing the objectives of the assignment can be loaded from CS 'Assess' and displayed to the student. If the message is blank, this step is skipped. The model then loads the appropriate information about the assessment from CS 'Assess' into a CS 'Dynass', which is used in the creation of a dynamic assessment form that displays the problem statement and assessment questions. The current student response is captured in CS 'Outcomes' and this is permanently stored in CS 'Output'. The student response is compared with the author's correct and diagnosed incorrect results, and the appropriate feedback is extracted from CS 'Assess' and displayed to the student. If the student's response is correct, the engine moves on to the next question. If incorrect, a new question name can be loaded from CS 'Assess' or the same question repeated, depending on the author's specifications.

To design a new problem, the author only needs to modify the three condition sets shown in the top left portion of Figure 1. The model structure remains unchanged, allowing the author to focus on the problem at hand. The procedure for designing a new problem would involve the following steps:

- Lay out the solution to the problem using the mathematical tools available within CAPE, including functions that can be written using the Python computer language.
- Identify errors that students are likely to make as they progress to the final result. Compute erroneous answers based on these potential mistakes.
- Decide on input variables (randomize vs. fixed) and units
- Design a question or sequence of questions that logically presents the problem to students, providing input information and appropriate units for each.
- Label the correct answer or set of answers, and a tolerance for each answer.
- Label diagnosed incorrect answers and tolerances.
- Construct meaningful feedback messages for each diagnosed response and for undiagnosed responses. Progressive remediation is supported, so a different message can be displayed after each iteration.
- Decide the number of times a student is allowed to repeat each question.
- Select a grade for each iteration of each question (if grading desired).
- Decide which resources, such as graphs, text references, www links, etc. you will provide the student

## Results

Figure 3 shows an example of a dimensional analysis problem statement authored with the design pattern. This is the fourth question in a sequence designed to teach students a procedure for identifying relevant dimensionless groups using the Buckingham Pi method. Additional information describing the method is available to the student if they press the "?" button on the eLMS interface. This particular question is a multiple choice, multiple answer assessment. Other assessment types supported by CAPE are fill in the blank (string, essay, float, integer), multiple choice (single answer mode), drop down lists (single or multiple answer modes), and True/False. For the problem in Figure 3a, more than one set of variables would be considered correct. The set selected is incorrect, and the diagnostic feedback is shown in Figure 3b. Based on this feedback, the student reduced the number of core variables from four to three by removing fluid density from the core, but the variables selected are still incorrect and additional diagnostics explain why the student's set of variables is incorrect (Fig 3c).

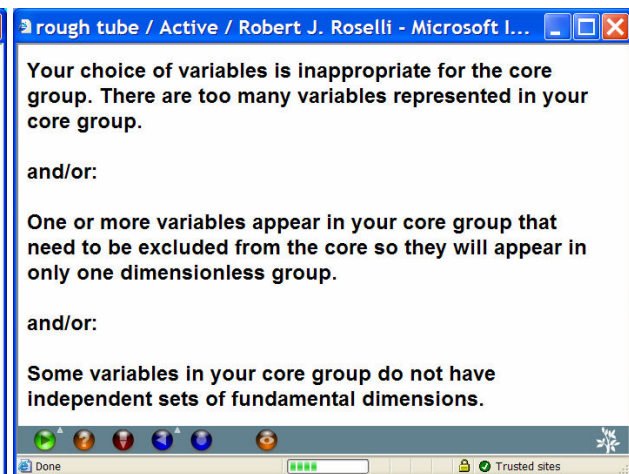
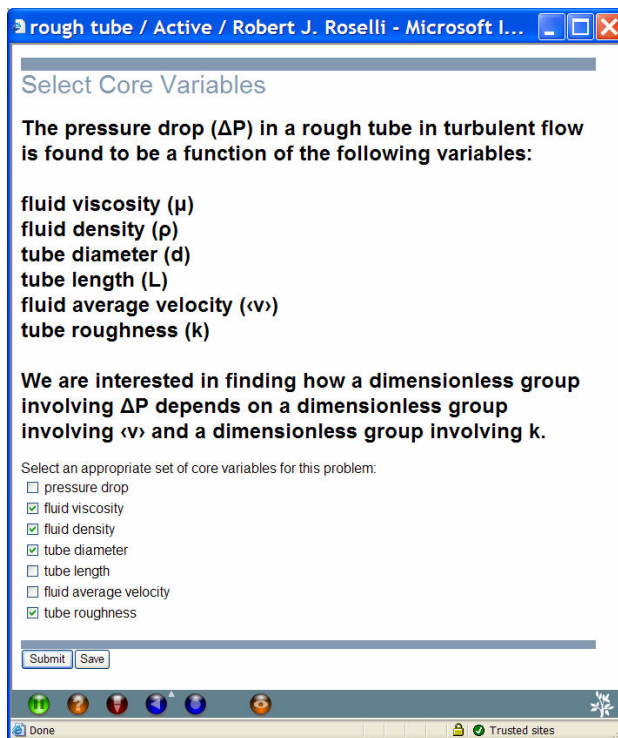
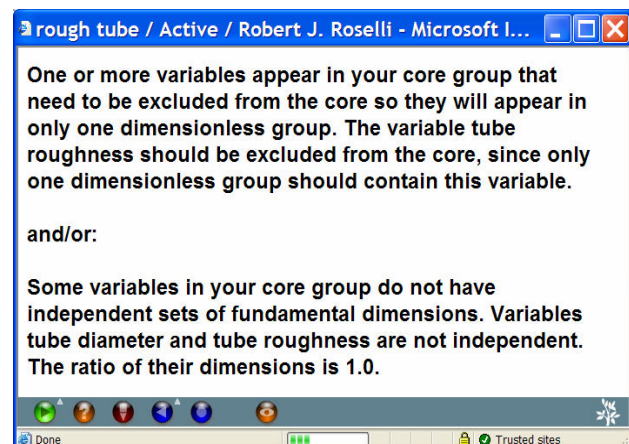


Figure 3a (left). Question and student response for selection of a set of core variables.

Figure 3b (above). Diagnostics for student response in Figure 3a.

Figure 3c (below). Diagnostics following elimination of fluid density in Figure 3a.

If the student fails to select a valid set of core variables on the third attempt, the module displays a valid set and explains why the student's set is not appropriate. Otherwise it uses the student's set for the remainder of the exercise. Before constructing the dimensionless groups, the module checks to make sure the student has the correct dimensions for each variable.



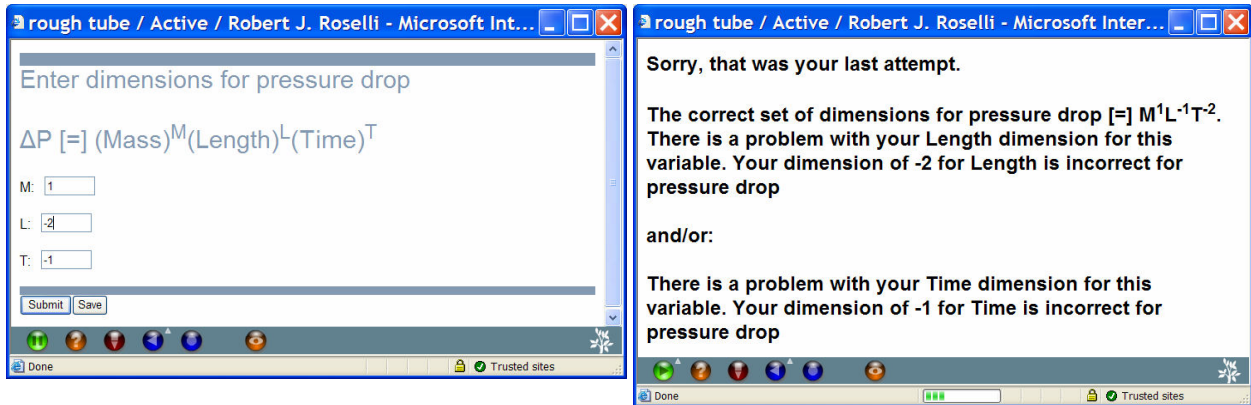


Figure 4. Question (left) and diagnostics (right)

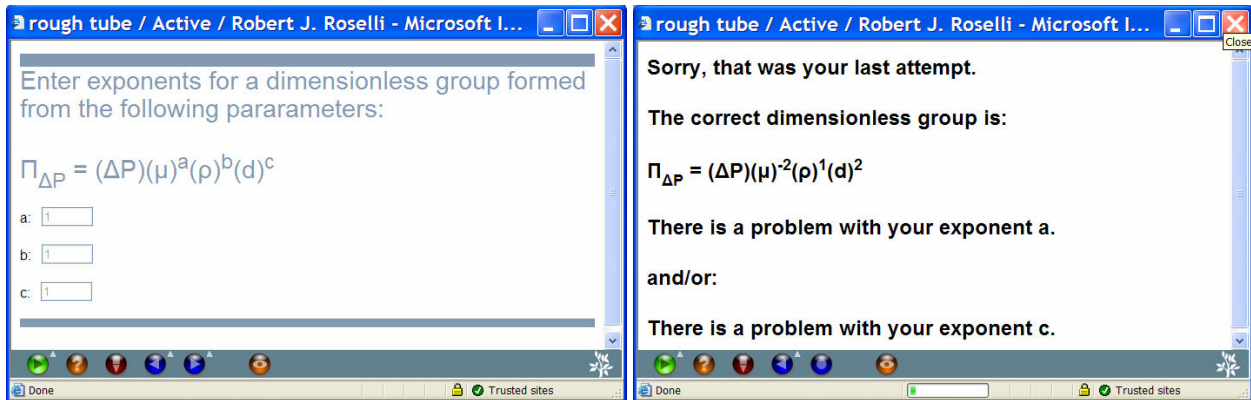
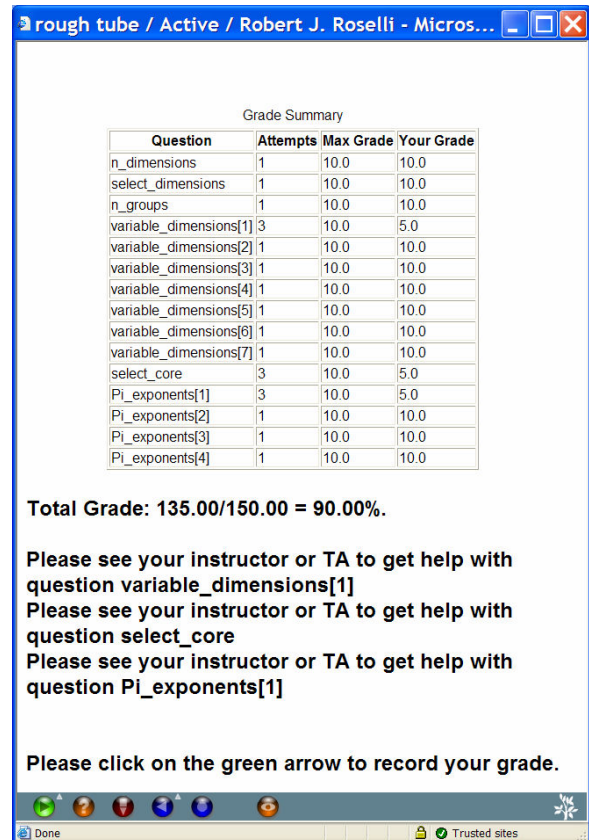


Figure 5. Question (above left) and diagnostics (above right).

Figure 6 (right). Grade summary.



An example of the question and feedback after the third attempt is shown in Figure 4. Figure 5 shows the question for constructing one of the dimensionless groups and feedback for an incorrect response on the final attempt. The grade break-down is displayed in Figure 6. The student is asked to seek help on three of the responses.

A second example dealing with the conduction of heat through skin is shown in Figure 7. This problem uses exactly the same CAPE design pattern, but with different data in the three condition sets Assess, Input, and Compute (Figure 1). In this problem, students must identify the correct equation to use and to identify what parameters are needed to compute the metabolic rate, choices not dictated by the

problem statement. A portion of the dialog is provided in the figure. Student answers are parsed to search for key words such as 'conductivity', 'area', or 'thickness.'

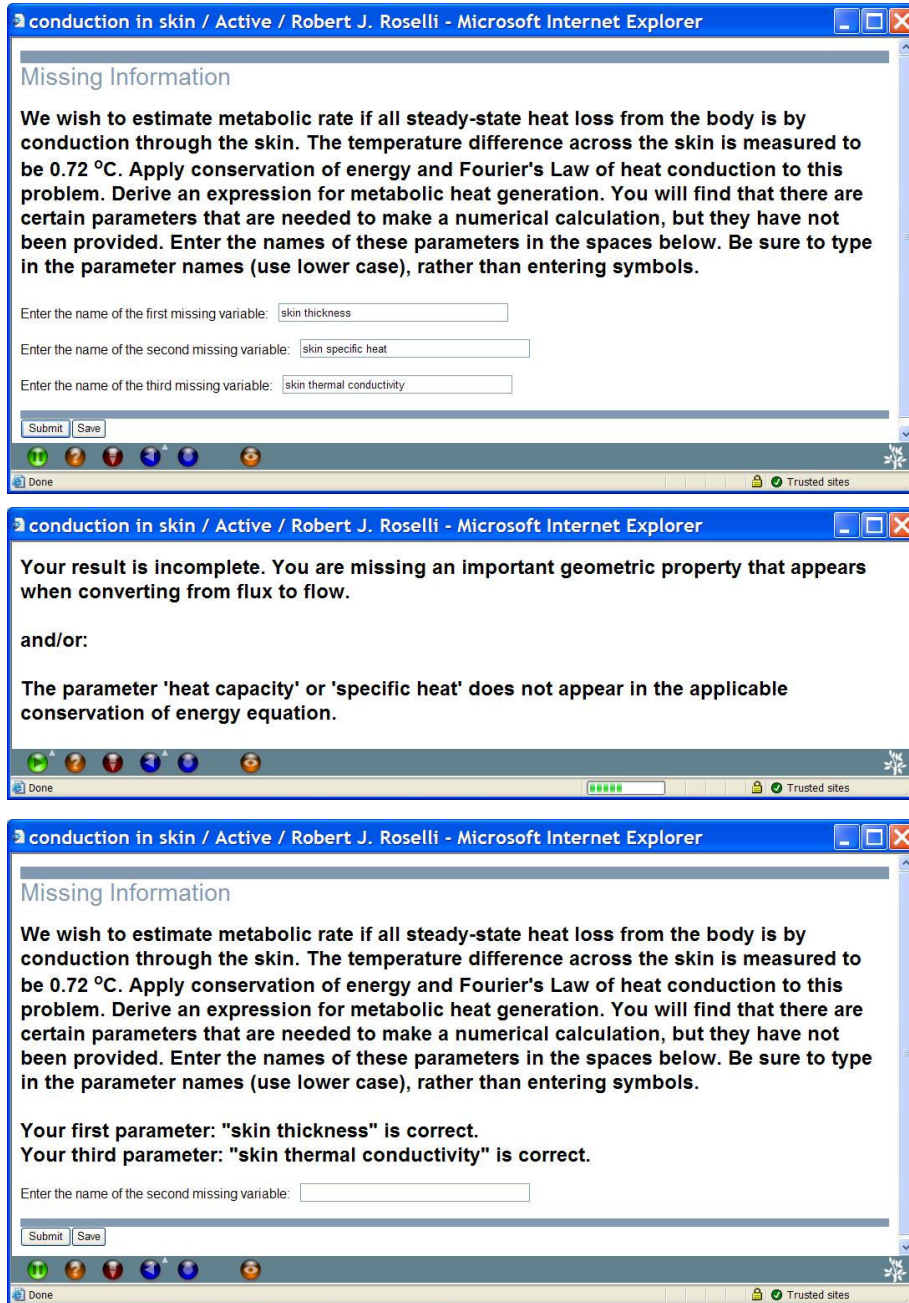


Figure 7. Portion of dialog in Conduction in Skin module.

## **Discussion and Conclusion**

In both examples there is no single correct answer. The 'correct' set of dimensionless groups depends on the student's selection for the core variables. The correct metabolic rate depends on the student's estimate of the skin thermal conductivity, surface area and thickness. Students are asked to provide information at least 15 times in each example. If any of their responses are incorrect, they receive feedback that may help them understand what they may have done wrong. The author has complete control over this feedback, which can become progressively more helpful with each incorrect response. Authors can provide students with intermediate results if they cannot answer a question correctly within the prescribed number of iterations. The author also has control over whether to grade the problem or not, and if so, how many points are associated with each part and with each missed response.

Since only the data in three condition sets are altered for each new problem, not the CAPE design pattern, development time should be significantly reduced. Furthermore, errors in design logic are eliminated. New diagnostic messages can be easily added to an existing module, without any change to the design pattern. In addition, problems can be easily broken down into smaller segments, if desired, by simply adding new questions to an existing data set. Again, the design pattern remains unchanged.

Students generally like these assignments because of the diagnostic feedback and instant grading. However, they also like the ability to go back over the problem in 'review mode,' either by themselves or with the instructor to discover where they may have gone wrong or to review material before an examination.

In conclusion, we believe that this reusable design pattern will: 1) significantly reduce faculty design time for constructing diagnostic modules, 2) eliminate logic errors likely to accompany new design patterns, 3) allow easy modification of existing modules, 4) allow TAs to design modules, and 5) improve student performance by providing meaningful diagnostics.

## **Acknowledgement**

This work was supported primarily by the Engineering Research Center Program of the National Science Foundation under Award Number EEC9876363.

## **Bibliography**

1. Howard, L. Adaptive learning technologies for bioengineering education. *IEEE Engineering in Medicine and Biology Magazine* 22:58-65, 2003.
2. Roselli, RJ and Howard, LP. Development of online homework problems that provide instant feedback and remediation to students. Annual BMES Conference, Nashville, TN (CD ROM, Omnipress), 2003.
3. Troy, JB and Linsenmeier, RA. Optimizing the delivery of content in physiology instruction. *IEEE Engineering in Medicine and Biology Magazine* 22:80-87, 2003.
4. Giorgio, T.D., Brophy, S.P., Birol, G., McKenna, A.F. and Smith, H.D. Assessment of Educational Modules Based on the "How People Learn" Framework Delivered to Biotechnology Learners at Two

Universities. *Proceedings of the Second Joint EMBS-BMES Conference* (CD-ROM, Omnipress): 2642-2643, 2002.

5. Roselli, RJ, Howard LP, Cinnamon, B, Brophy, SP, Norris, PR, Rothney, MP and Eggers, D. Integration of an Interactive Free Body Diagram Assistant with a Courseware Authoring Package and an Experimental Learning Management System. ASEE Annual Conference, (CD-ROM DEStech Publications) Session 2793: 10 pages, 2003.