Interactive Problem Solving for Mechanical Engineering on the World-Wide-Web

Gregory G. Kremer and Bhavin V. Mehta
Ohio University

Abstract

The world-wide-web has established its place in computer-based education due to its great advantages in terms of access, distribution, communication, and timeliness of feedback. However, software for internet-based applications is still far behind that for PC based applications in many respects, most notably in the types of interactive experiences available and in techniques for dealing with mathematical equations and symbols. Researchers at Ohio University are developing a web-based Interactive Problem Solver (IPS) with numerous Java applets and applications that attempt to address some of the limitations of existing web-based educational tools, especially with regards to interactivity. Rather than merely allowing students to change parameters and see the effect on a system response without ever having to "get their hands dirty" developing and solving the equations, the IPS attempts to implement the key features of an intelligent tutoring system (active learning, forced reflection, targeted feedback) by requiring "unguided" student inputs and returning instant formative feedback in both textual and visual forms. The IPS is a student-centered environment for learning Dynamics on the world-wide-web in which the student controls various details of the problems that will be solved, creates free body diagrams by pointing and clicking to select systems and to place forces, enters symbolic equations representing the mathematical model, solves the mathematical model using available computational software, and evaluates the physical realism of the solution. Students are given immediate feedback in direct response to their inputs at every step in the problem solution, but rather than being given the solution they are given hints for discovering what they did wrong and how to correct it. Additionally, tools for assessing student learning and for assessing the impact of the IPS on student learning are integrated within the IPS.

I. Introduction

The most appropriate method of presenting material in an engineering course depends on many factors, but two of the main considerations are the intellectual maturity level of the students and the desired objectives or outcomes of the course. Although students in entry-level mechanical engineering courses have been through the Calculus and Physics sequence, they are often unprepared for the "new thinking" required to solve engineering analysis problems. The process of reading a description of a physical situation, deciding which analytical theory applies, converting the physical situation into a solvable mathematical model, solving the model, and finally visualizing the forces and motions to evaluate the physical realism of the solution can be a daunting task. It was theorized that an educational tool that provided a problem-solving framework would be a useful educational supplement for helping entry-level engineering students develop the skills and the mindset for solving engineering analysis problems. Details of the development of that framework can be found elsewhere.[1,2] For practical purposes, it was determined that the framework would be most useful if it was implemented in a form similar to an intelligent tutoring system but instead of in PC based software like most tutoring systems to do it on the world-wide-web. This paper reviews the ongoing effort at Ohio University to implement a problem-solving framework and related pedagogical tools in a web-based Interactive Problem Solver (IPS) that can be used as a supplement in a sophomore-level Dynamics course to increase student learning.

The IPS is a student-centered environment for learning Dynamics on the world-wide-web in which the student controls various details of the problems that will be solved, creates free body diagrams by pointing and clicking to select systems and to place forces, enters symbolic equations representing the mathematical model, solves the mathematical model using available computational software, and evaluates the physical realism of the solution. Students are given immediate feedback in direct response to their inputs at every step in the problem solution, but rather than being given the solution they are given hints for discovering what they did wrong and how to correct it.

II. Educational Tools and Interactivity on the World-Wide-Web

The world-wide-web has established its place in computer-based education due to its great advantages in terms of access, distribution, communication, and timeliness of feedback.[1] Additional features of web-based applications that are likely to be beneficial if used correctly but for which there is not yet sufficient proof of their effectiveness include building a connected learning community that enables collaboration, creating educational environments, providing a hyperlearning experience, and providing an interactive learning experience.[3]

There is significant interest in developing both computer-assisted instructional modules and web-based instruction. The National Science Foundation Division of Undergraduate Education (DUE) has recently sponsored research in web-based instruction, for example, "An Internet-Based Interactive Learning System for Computing Science", "Electronic Homework and Intelligent Tutoring on the World Wide Web: Course Delivery Tools for Large Enrollment Classes", and "Interactive Visualization via Java for Signals and Systems".[4] Additionally, the NSF DUE has supported projects to develop computer-assisted instructional modules for entry-level mechanical engineering courses, including "A Learning Superstructure for BEST Dynamics Software", "Interactive, Conceptually Based Multimedia Instruction for Introductory Mechanics", and "Computer-Based Learning Aids for Problem Solving in Mechanics of Materials".[5] Other notable examples of educational tools and interactivity on the world-wide-web include recent Premier Courseware of 2000 award winner "Project Links: Interactive Web-Based Modules For Teaching Engineering",[6] and PIVoT (The Physics Interactive Video Tutor).[7]

Interactivity is one of the current buzzwords in multimedia and internet-based software applications, and everyone seems to agree that interactivity is a key feature in an effective computer-based learning environment. However, our review of a large number of educational materials (those accessible on the web or via CD) showed that there is a wide disparity in the types of activities that are called interactive, and there is a lack of research on determining the impact of various forms of interactivity on student learning. When we speak of person-to-person interaction we are normally referring to a 2-way discussion or exchange of information in which one person's response is dependent upon the other person's question (input/output relationship).

On-line interactivity can mean students interacting with content (man-to-machine), students interacting (synchronously or asynchronously) with the instructor (dialogue), and students interacting with other students (in a forum discussion or working collaboratively). Figure 1 categorizes the different activities or features of educational software that are commonly called interactive and details some of their benefits.

Control of Information Flow (Content Interactive)

With hypertext/hyperlearning, the site organizes and presents information similar to a textbook but with infinitely more resources (those of the internet) easily available. The student's choice of when to click on resources for more information is the primary form of interactivity, with the benefit of allowing students to direct the pace and order of their studies.

Communication/Collaboration

Students are linked via chat (real-time dialogue) or email (asynchronous communication) with peers or instructor, various file sharing tools are available for collaboration, and files can be submitted electronically for off-line review by the instructor. The interactivity here follows the person-to-person dialogue model, and the benefits include creating a community of learners and enabling timely communication and feedback.

Exploration of the effects of parameters on solutions

Interactive examples allow students to input/modify key parameters and view the change in the solution (often making use of visualization software that provides animation capability) to get a better "feel" for the system being analyzed. Although some sites show the equations used to get the solution, few if any currently require the student to help generate the solution (analysis/problem solving is transparent).

Interactive Customization

Interactive customization provides a personalized WWW experience using a database that is dynamically updated with the user's latest preferences and actions based on data continuously collected from the website. The site attempts to change in real time to better suit the user. So far, this capability is found mostly in business sites with the goal of increasing sales

Figure 1: Categories of interactivity

The difference between active learning and passive learning ultimately depends on the responsiveness of the learning environment. The different categories shown in Figure 1 represent vastly different levels of interactivity, and there is significant debate over whether or not these levels of interactivity create a truly active learning environment. For example, in the common interactive exercise model a student selects a value for a parameter and the program computes and outputs the results for the student to observe. For most students these interactive exercises are still passive learning activities since the solution method is transparent and requires no student effort. The student sits back and lets the computer do the work, and the result is similar to a lecture presentation model wherein information created by an expert is shown to the student.

Proceedings of the 2001 American Society for Engineering Education Annual Conference & Exposition

Increased student involvement, interpretation of student input and targeted feedback are the major qualities that seem to be missing in web-based educational tools, likely because these qualities are very expensive to provide. Based on pedagogical research, interactivity on the level of a one-on-one tutoring session is most desirable, with the student being guided towards a correct solution based on his/her incorrect attempts to solve the problem rather than just being shown the solution without having to invest any effort in generating it. For a detailed discussion of the problem-based learning and learning environment research that serves as a background for this paradigm, see the recent ASEE conference papers "A Web-based Interactive Problem Solver for Enhancing Learning in Engineering Mechanics"[1] and "Interactive Dynamics: A Collaborative Approach to Learning Undergraduate Dynamics.[8]" One key point from this research is that students given one-on-one tutoring from an instructor have far superior final achievement compared to students in a conventional teaching environment. Because of its proven effectiveness, many computer-based learning environments have attempted to replicate the tutoring experience, resulting in an entire category of intelligent tutoring systems (ITS). Unfortunately, to the authors' knowledge none of these ITS systems are web-based. ITS and Computer-Assisted Instruction research indicates that the key factors in the success of personal tutoring are reinforcement, encouragement, constant feedback, and a forced process of reflection and correction.[9] These are the factors that we are trying to implement on the world-wide-web in our interactive problem solver.

III. Distinguishing Features of the Interactive Problem Solver (IPS)

Taking the global view, student learning depends on a proper learning environment. Interactive experiences are important but must be viewed as tools operating within an overall learning environment. As previously discussed, the learning environment in the IPS is based on a problem-solving framework, and the interactive tools attempt to replicate a tutoring experience. Figure 2 details the tutoring system logic for an individual step within the IPS framework. Note that this is the goal for the software, it has not yet been completely realized in version 1 of the IPS. The IPS creates a log file of all student responses which can be viewed at any time. This log file can be used for an overall evaluation of the students performance (the number of wrong answers and/or the number of unanswered questions) for a particular problem and can be used to determine if student performance is acceptable (proceed to next problem) or unacceptable (must complete another similar problem or visit with the instructor before proceeding).

The IPS uses realistic vehicle dynamics problems selected specifically to motivate and drive the learning, with the goal of developing each learning module (series of related problems) such that it shows the connection of Engineering Mechanics with background material and advanced topics (such as machine design) for further motivation. The URL for the IPS website is http://www.ent.ohiou.edu/~dynamics/. Although the site is under construction, guests to the site can review a demo to better understand our plans and the site's features. The remainder of this section includes screen shots of some of the features and Java applets used in site, along with short textual descriptions.

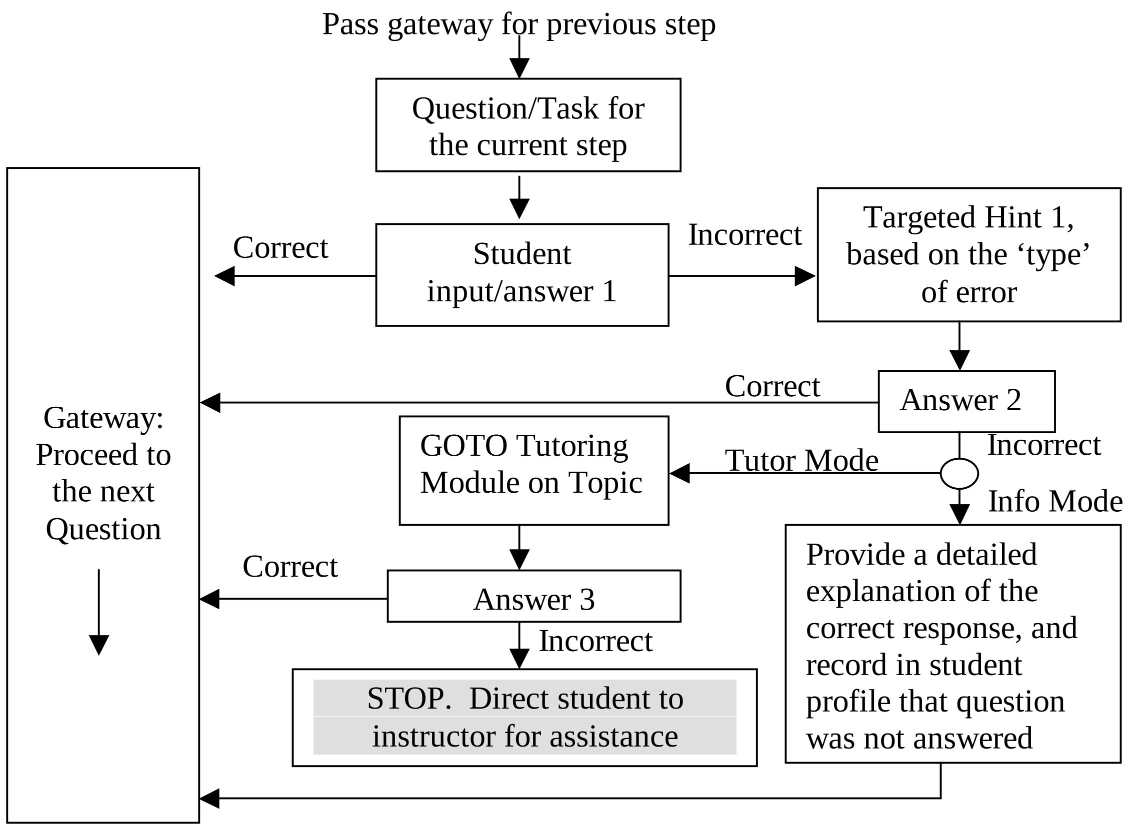

Figure 2: Tutoring System logic for a specific step within the IPS

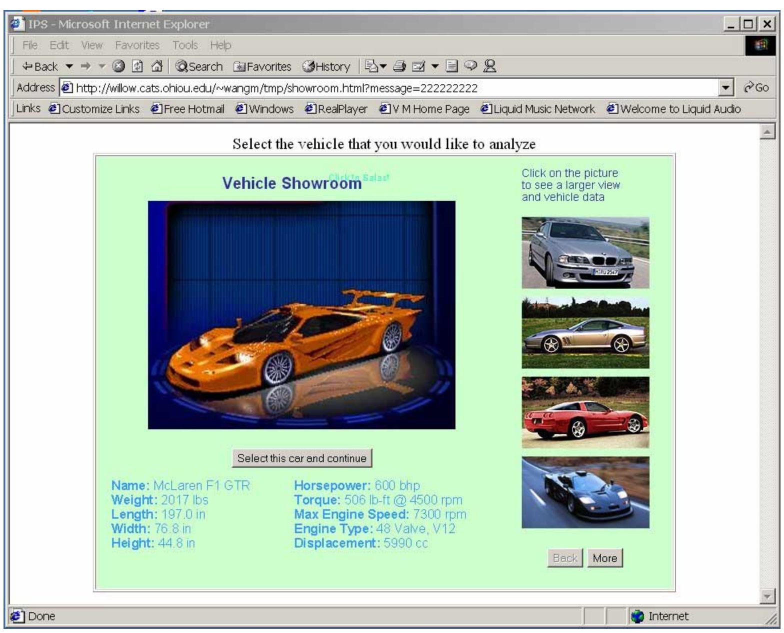

Figure 3: Control over problem selection.

Increases student involvement, motivation, and can be controlled to ensure that all students end up working slightly different problems to discourage copying.

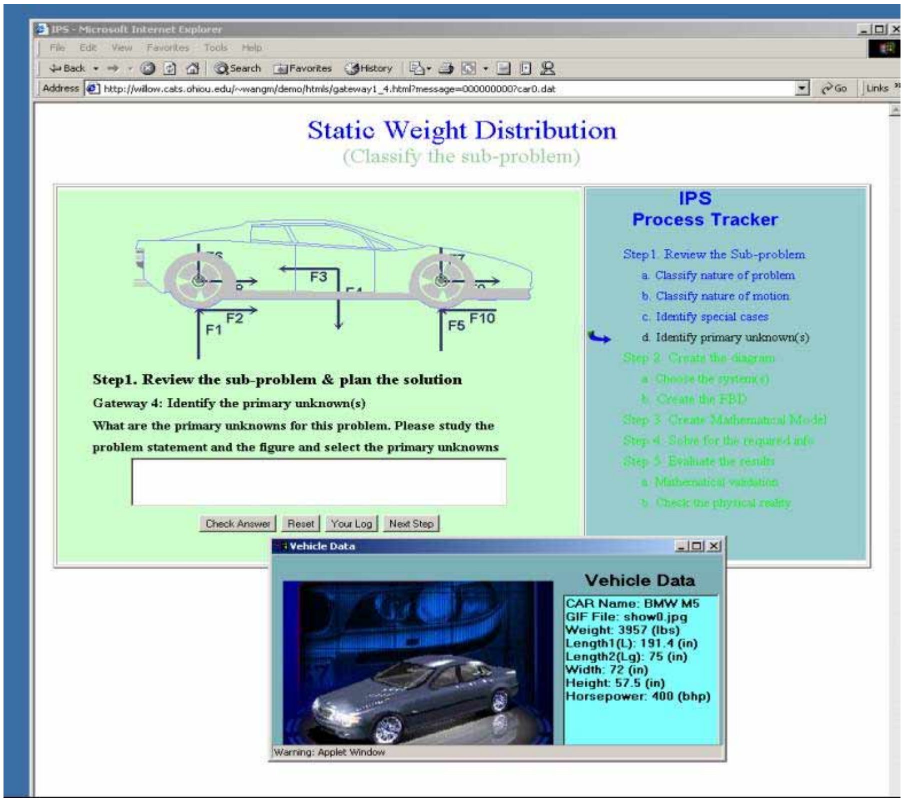

Figure 4: Multiple Screens, showing framework, problem data, input/feedback

The IPS tracker always shows the framework outline on the right of the screen so the student is aware of the overall solution process. A separate screen showing the problem data is always available. Mouse click input and text box instant feedback modes are shown.

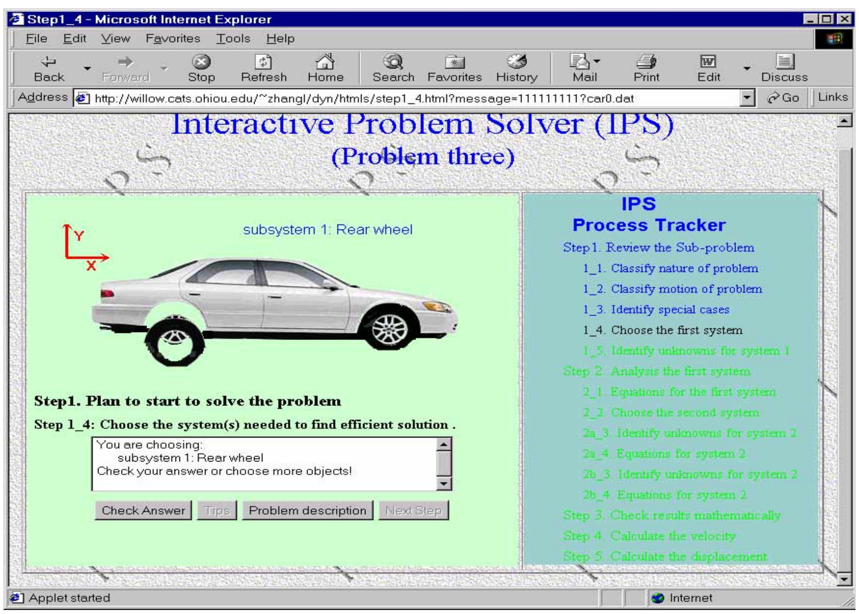

Figure 5: Selecting the "active system" for the FBD

Many different systems are valid, the IPS allows any system selection that will lead to an efficient solution.

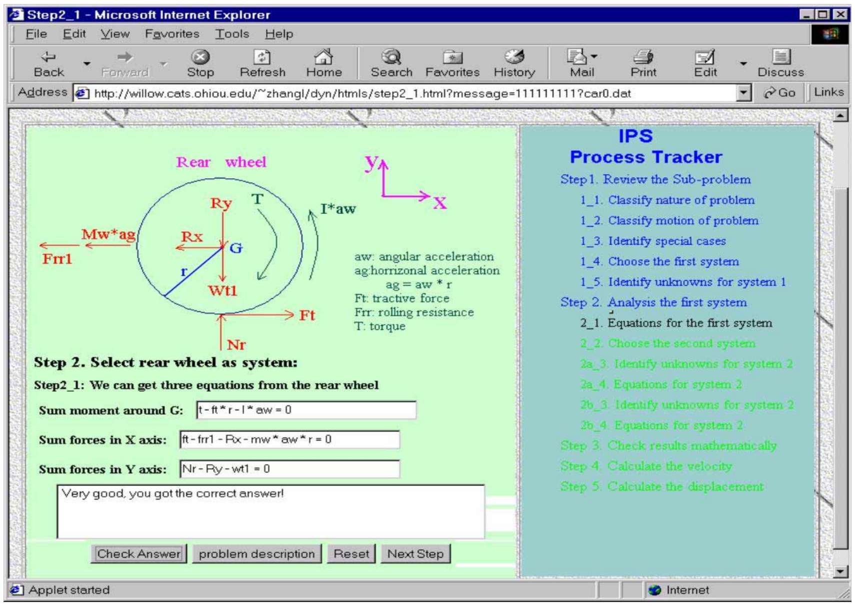

Figure 6: Student input of symbolic equations of motion

The system of equations is evaluated for correctness before the student proceeds.

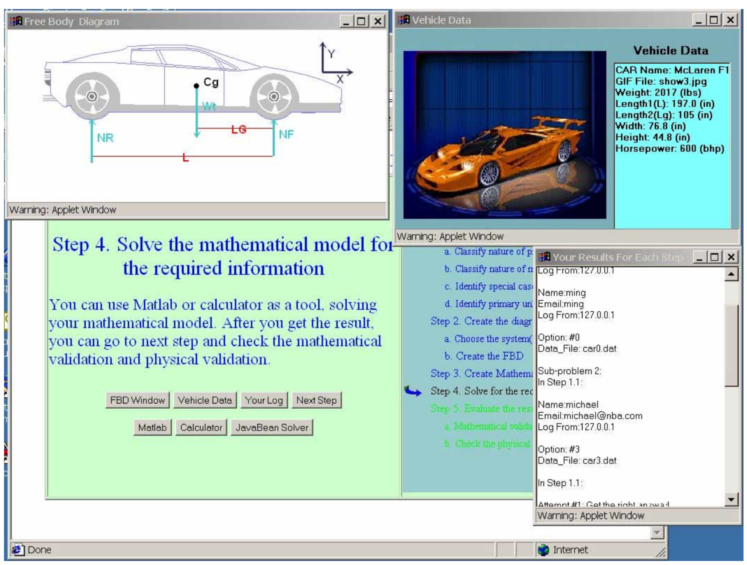

Figure 7: Computational tools, the log file, and the FBD screen

The student selects an on-line or off-line computational tool to solve the system of equations. The additional screens are always available so the student has access to all necessary information.

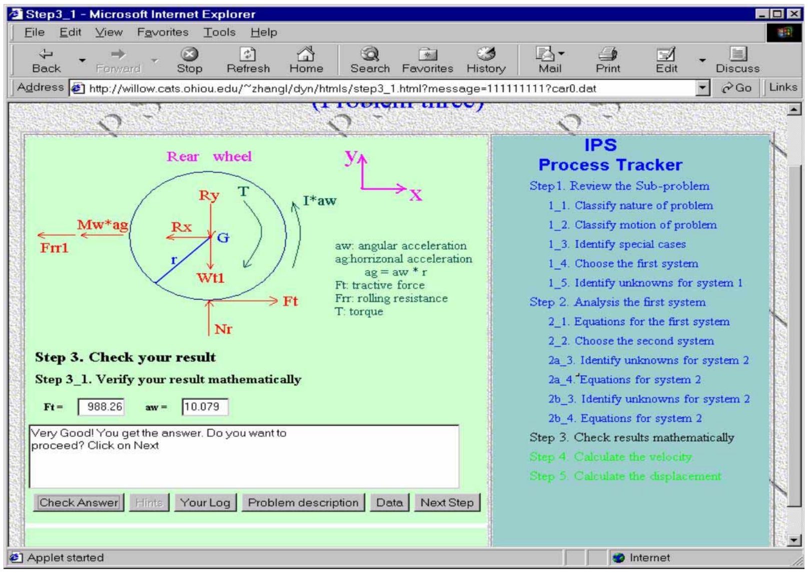

Figure 8: Input and evaluation of the numerical result

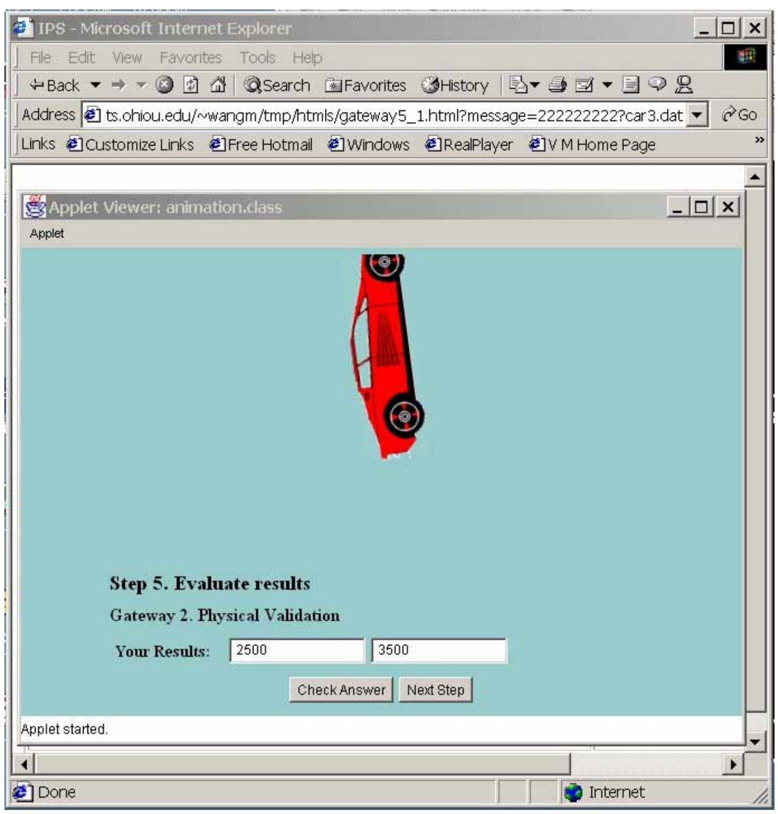

Figure 9: Simulations for Physical Validation of Results

A very important step in the solution process in which the student's input is simulated or graphically shown and the student must evaluate and accept the result or retreat to an earlier stage of the solution and rework the problem.

IV. Conclusions

Various forms of assessment have been carried out for the IPS, including on-line evaluations of usability and usefulness of beta versions of the site. Results of the on-line surveys showed identified numerous usability issues, but were generally positive and encouraging with respect to usefulness. Based upon student comments and our experience to date in using a problem-solving framework off-line in a Dynamics course, the original IPS framework will likely be simplified and clarified. The five main steps in the revised Dynamics problem solving framework are 1) Define problem, 2) Create diagrams, 3) Create mathematical model, 4) Solve mathematical model, 5) Check physical realism of result. We are confident that revision two of the IPS will be more usable and be even better accepted by students.

Based on our experiences, we offer the following lessons learned. In general, it is very important to keep educational tools very focused and as simple to use as possible to avoid confusing students with software or other issues not directly related to the educational objective. In other words, start out as simple as possible and only add complexity once the simple system is operating as desired. Creating web-based educational tools is very expensive in terms of initial time commitment. The tools do offer potential long term time savings and are consistent with the emphasis on enabling distance education, but the initial start-up requires significant time and funding to accomplish properly.

Our effort in developing the IPS continues but has been slowed recently by lack of funding. Therefore, future plans depend largely on our ability to find partners and funding to continue the project. Ultimately, we would like to extend the IPS to include more topics and tasks both in Engineering Mechanics and throughout the curriculum to create a "web-based curriculum" supplement. Upper-level students would be given the freedom and responsibility of semi-autonomous learning, while entry-level students could rely on the structured format and the availability of guidance and feedback.

Bibliography

1. Kremer, G., Pasic, H., and Mehta, B., "A Web-based Interactive Problem Solver for Enhancing Learning in Engineering Mechanics," *Proceedings of the 2000 American Society for Engineering Education Annual Conference & Exposition*, Session 1368, 2000.
2. Kremer, G., "Teaching a Rigorous Problem-Solving Framework in Entry-Level Mechanical Engineering Courses – Theory and Practice," *Proceedings of the 2001 American Society for Engineering Education Annual Conference & Exposition*, Session ####, 2001.
3. Ausserhofer, A., "Web-based teaching and learning: A panacea?," *IEEE Communications Magazine*, V. 37, N. 3, pp. 92-96, 1999; Bilotta, E., Fiorito, M., Iovane, D., and Pantano, P., "An educational Environment using WWW," *Computer Networks and ISDN Systems*, V. 27, pp.905-909, 1995; Coppula, D., "Getting Hyper over Learning," *ASEE Prism*, pp. 12-17, May-June 1997.
4. Palakal, 1996, Day, 1997, and Rugh, 1999, as documented on NSF DUE Project Information Resource System (http://www.her.nsf.gov/PIRstart/.
5. Flori, 1996, Bunge, 1995, and Steif, 1999, as documented on NSF DUE Project Information Resource System (http://www.her.nsf.gov/PIRstart/.
6. Holmes et al., "Project Links, Interactive Web-based Modules for Teaching Engineering," http://links.math.rpi.edu/, 2000.
7. Larson, R. "Interactive Physics Tutor Personalizes Help for Students," Syllabus, November 2000, p. 46.

8. Gray, G. and Costanzo, F., "Interactive Dynamics: A Collaborative Approach to Learning Undergraduate Dynamics," *Proceedings of the 2000 American Society for Engineering Education Annual Conference & Exposition*, Session 3268, 2000.
9. Shute, V.J., and Regian, W., "Principles for Evaluating Intelligent Tutoring Systems", *Journal of Artificial Intelligence and Education*, 4(2/3), 245-271,1993; Psotka, Intelligent Tutoring Systems, 1988; Venezky, 1991.

GREGORY G. KREMER
Gregory Kremer is an Assistant Professor of Mechanical Engineering at Ohio University. His area of expertise is Mechanical Systems Design, and he has extensive industrial experience in this area. Dr. Kremer received his B.S. degree in Mechanical Engineering from Rose-Hulman Institute of Technology in 1989 and his Ph.D. degree in Mechanical Engineering from the University of Cincinnati in 1998.

BHAVIN V. MEHTA
Bhavin Mehta is an Associate Professor of Mechanical Engineering at Ohio University. His areas of expertise include CAD/CAE/CIM, artificial intelligence and virtual reality, and web-based implementations. Dr. Mehta received his Masters degree in Chemical Engineering from Ohio University in 1988 and his Ph.D. in Interdisciplinary Engineering from Ohio University in 1992.