Session 2257

Using the Internet to Support Problem Based Learning

Louis J. Plebani, Joseph C. Hartman
Lehigh University

Abstract
This paper reports on the use of the Internet to support problem-based learning, a trend in pedagogy that is used to engage students in learning by presenting them with problems they perceive as more realistic than textbook problems and by requiring them to fill in gaps when presented with a situation they do not readily understand. This past fall, we implemented a relatively simple homework/project using the Internet in our senior level “Production and Inventory Control” course in the Department of Industrial and Manufacturing Systems Engineering at Lehigh University. This paper discusses an overview of the problem/learning tool implementation and student reaction.

I. Background

Teaching mathematically oriented subjects such as basic statistics and economic decision analysis is often frustrating. At their heart, these subjects are not difficult (after all, they are by definition logical and for the most part follow consistent rules), but many students have difficulty because they are not motivated to drill the homework problems required to cement the concepts taught in lectures. At the same time, it is widely accepted that students across the educational spectrum understand material better, retain it longer, and enjoy their classes most when they take the lead to think about what they are doing. Research [1,2] has shown that there are two major motivators in this regard: (1) when students work on problems they perceive as meaningful or relevant; and (2) when students are placed in a competitive situation in the role of a problem solver confronted with an ill-structured problem.

In order to provide relevance, an instructor in quantitative courses often introduces anecdotal evidence concerning the application of the basic concepts. This may succeed in capturing interest during a class period, but it is not often possible to provide problems that piggyback on the anecdotes. Usually, if a synergistic exercise is provided, it is in the form of a project because simple homework problems do not permit the complexity required to truly stimulate deep thinking about relevant techniques, solution tradeoffs and their consequences. Textbooks are seldom of much help as they are usually filled with end of chapter exercises that are anything but ill structured. Many students typically do homework by perusing the relevant chapter until they find an example that looks like the current exercise. They then substitute data into the example framework and get an answer without appreciating any details of the method or its application. The downside of projects is the large amount of instructor time that is consumed in developing, managing and grading a realistic project. The effect of this, rightly or not, is that project type problems tend to be excluded from the realm of drill exercises that are often required in science and mathematics.

In an attempt to solve this dilemma of too little learning on the student side and too much tedious work on the instructor side, we have begun the development of a web-based system to bridge the gap between drill exercises and term projects. The system will allow instructors (without excessive development effort) to present problem scenarios to students which would be based on discrepant events, incongruities, anomalies, or stated needs of a client. This simply means that the students have to gather information as they engage in the problem-solving process. This may include data from the present situation as well as information based on prior knowledge or resource searches. Once they have accumulated necessary elements, students are free to formulate and test tentative hypotheses "off-line". When students enter a solution, they receive immediate feedback in the form of a score based on the quality of their solution and resources used in obtaining it. Students are free to do as many problems as they desire since each problem generated by the system is unique. Scores for the exercise are based on the average of all problems attempted.

II. Problem/System Overview

Generically, the operation of the system is as follows. A student/team requests a problem from the system via a web connection. In response to the request, the system creates and initializes a unique problem scenario. This problem scenario conforms to a problem template that consists of a problem story, generated problem parameters, and a response/evaluation environment. The problem story is constant for each scenario generated from a particular template and describes a problem to be solved. The problem parameters are pseudo-randomly generated for each problem using problem specific algorithms. The response/evaluation environment is unique for each generated scenario and is the virtual laboratory through which students obtain problem specific data and within which their solutions are evaluated. The environment contains the essence of real problem environment characteristics, most notably that data is subject to error and that data is acquired at a cost. Student interactions need not conclude in a single session. A student may login/logout any number of times before submitting a solution. They are free to formulate a solution using any means available to them, be it analytical or simulation. Upon solution submission, results of the evaluation are returned as a score based on template specific rules that may or may not be completely known in advance to the problem solvers. Both solution accuracy as well as the resources used in achieving the solution, e.g., the number of data points requested, calendar time, etc. can influence the final economic score. The system database stores all problem-related activities for each student/team and facilitates competition between students/teams.

III. Prototype Implementation

A prototype problem was piloted in our undergraduate "Production and Inventory Control" course. In order to implement a system as described, server-side processing is necessary. Because of the privileges required to do server-side processing it was most practical for us to implement a web server on one of our own computers rather than using one maintained by the university. We used a desktop PC running the Microsoft Windows NT Server operating system and Microsoft Internet Information Server for the web server. Since the number of users was small by web standards, Microsoft Access served as the database system. The use of Microsoft products was primarily because of convenience rather than any processing superiority. Freeware such as the Linux operating system and the Apache web server could also have been used.

When the students accessed the problem web site, they were required to login in the routine manner of entering a username and password. The system used the username to uniquely identify all entries in the database. Once logged in, students were presented with the problem story and action form shown in Figure 1. All constants were uniquely generated for each problem. The story described a continuous review inventory situation that included ordering costs, holding costs and shortage costs with stochastic demand and lead times. The problem required students to determine a (Q, R) policy. Demand was not specified and had to be determined from samples acquired at a cost.

In order to obtain demand data, the students filled in the number of demand samples desired on the action form and submitted the request. Data was randomly generated, based on the stored problem specific parameters, and displayed along with summary statistics (see Figure 2 for a depiction). After digesting the sampled data, the student could (1) request more data, (2) enter a solution, or (3) exit the system to return later. Students could leave the system, in order to analyze and ponder, any number of times for any length of time without any negative impact on their score. Since all initial and generated information was retained for each individual student in a database, the system returned to the same state (as if the student had not left) when the student re-entered the system by logging in again. Students could review all data at any time.

Once students completed their analysis they entered values of Q and R and submitted the solution for evaluation. For this problem, a simulation over a one-year period was used for the evaluation. The re-order point was used as the initial inventory and daily demand was generated for one year with orders of size Q placed whenever the inventory position reached R. Over the course of the one-year simulation, ordering costs, holding costs, and penalty costs due to backlogged and/or lost orders were tallied. These costs, in addition to a cost for attaining sample data, were then totaled into a final cost value. A visual output of the simulation along with problem parameters and the final cost figures were presented on an evaluation page (see Figure 3).

The evaluation page also contained an economic score that was assigned to the solution and stored in the database. The development of a fair method of determining an economic score is both important and a non-trivial part of problem scenario development. Above all, students must perceive the system as evaluating them fairly. This would not be as difficult if all students received an identical problem, however, an important aspect of the system is that students always receive a unique problem.

For the prototype problem, we settled on the following evaluation method that provided a reasonable element of fairness and did not require an unreasonable amount of server processing time. To prepare for problem evaluation and scoring we developed a methodology to assure that generated problem parameters maintained reasonable pairwise relationships and did not cause very unusual optimum such as an order-on-demand solution. This was accomplished by using stochastic algorithms that bounded parameters to specific ranges and enforced reasonable relationships. For example, once item costs were generated, holding costs were generated as uniformly random 10 to 20 percent of item costs; time dependent stock out costs were generated as uniformly random 5 to 10 times item costs; and sampling costs were based on a percentage of total cost as calculated by a simple EOQ formula from previously generated values.

After student scores were evaluated, the scoring system calculated values of Q and R and the simulation was rerun using these values. Total costs were computed in a manner identical to the student evaluation with the exception that no sampling costs were included. The student economic score was calculated as 100·(student score)/(system score). In computing the system (Q, R) values, the system had the advantage of knowing the true probability distributions of demand and lead-time. However, an important constraint on the system was that the computation requirements of the solution method had to be kept to a minimum in order to produce rapid response to the student and more importantly not to bog down the server and influence overall system response. We used a variation of the heuristic treatment of the "fixed reorder quantity system with lost sales" model as developed in Johnson and Montgomery **[**3**]**. The variation used a fixed three step iteration and a fast executing polynomial approximation to the normal loss function. Since the system's solution did not include sampling cost, a score close to 100 was excellent. It should also be noted that as the system solution was obtained with a heuristic and that the solution was evaluated in a simulation, it was possible that a student could beat the "optimal" decision based on randomness in the system (emphasizing the real world nature of the problem and randomness).

The function of the economic score served two functions. First, it normalized scores among students such that individual problem data did not influence the final score. Second, it provided a mechanism in which students could compete as the economic scores could be ranked at the discretion of the instructor.

IV. Implementation Issues and Discussion

Upon completion of at least two iterations of solving the prototype problem, students were requested to submit an on-line evaluation of the system and make suggestions for improvements. The survey form is shown in Figure 4. Reaction to the system was overwhelmingly positive, as students cited numerous benefits to the system. Specifically, students noted that they were motivated to do the problems and the system was convenient and easy to use. Examination of the surveys suggested that the students were motivated to drill the problems because they were perceived as realistic and challenging and they were able to apply knowledge acquired in class. Also, the scoring rules drove them to continually improve. Here is a representative sampling of comments from students to that effect:

> "The problems were more realistic than problems I faced in the book. It was challenging because I learned to use the formulas and see how they worked in a more realistic environment."
>
> "The problem challenged me. At moments I was confused and had to rationalize my answers which made me understand the concepts better."
>
> "It challenged me because I had to decide what data was relevant to the problem."
>
> "The problems were realistic as well as the way they were graded. Even if the exact answer wasn't determined, an effective economic score could be determined, which is close to realism."
>
> "The problems did seem realistic and you have to weigh the option of how much you want to spend (on data sampling) compared to accuracy, as in real life."
>
> "I think this system would be good for homework instead of doing problems from the text. Text problems just apply a few equations, and people don't have to think. However, these problems make people think."

> "Taking notes, getting lectured and taking quizzes is such an unrealistic way of learning. Small projects like this are more in tune with reality because (a) I can work on it on my own time and (b) I can use any and every resource I can find."

> "Being presented with different situations kept the assignment fresh and challenging."

> "I would like a system like this because it involves competition. It would also motivate me to do more problems because if I did bad on a problem, I would want to try again and again to get better scores into my average."

A minority of students were less enthusiastic about the system. Some complained that they could guess answers as readily as compute answers. This was by design, as solution methodologies were not imposed on the students in this example. Here were some negative comments provided by students:

> "I messed up the first two problems, so I spent a lot of time trying to fix my average by doing more problems. At the conclusion of the exercise (11 problems attempted), I felt I did more to fix my grade than learn about inventory control."

> "I definitely did not enjoy using it because I didn't learn anything. The only thing that I was concerned about was to get my score up. It was more of a guessing game than a calculated result."

With these comments, students offered suggestions to improve the system for future implementations. These included:

1. Allowing a student to solve an "example" problem for practice;
2. Providing a "hints" function that would provide more insight to the problem at a cost, much like sampling;
3. Posting student scores (or an average) such that students would have a "target" to beat;
4. Providing links on the summary page to programs such as EXCEL;
5. Building integrated problem scenarios that build on each other.

We examined the student scores to determine if we could draw any conclusions as to whether learning occurred. We found that while students were only required to solve two problems, the average number of problems solved per student was 6.2. The average problem score for the 54 students for all problems was 55.6, but the average score for the last problem done by each student was 65 compared to the average score for the first problem of 24.5. Though minimal evidence, this suggests that the students learned and improved over their iterations with the system.

V. Conclusions and Future Work

Although we are in our initial stages of testing the prototype system and integrating it as a learning tool into our undergraduate curriculum, we are encouraged by our initial results. This initial prototyping and testing is part of a larger goal of making a general problem based learning system available to all educators, without requiring that they have computer expertise.

Bibliography

1. Bonwell C. C. and J.A. Eison, Active Learning: Creating Excitement in the Classroom. Washington, DC: George Washington University, 1991.
2. Brooks, J. G., and M.G. Brooks, The case for constructivist classrooms, Alexandria, VA: Association for Supervision and Curriculum Development, 1993.
3. Johnson, L.A., and D.C. Montgomery, *Operations Research in Production Planning, Scheduling, and Inventory Control*, John Wiley and Sons, 1974.

LOUIS J. PLEBANI is an Associate Professor of Industrial and Manufacturing Systems Engineering at Lehigh Unviersity. He received his Ph.D. in Industrial Engineering at Lehigh, after receiving his M.S. in Operations Research at American University and B.S. in Engineering Physics at Lehigh.

JOSEPH C. HARTMAN is an Assistant Professor of Industrial and Manufacturing Systems Engineering at Lehigh University. He received his Ph.D. and M.S. in Industrial Engineering from the Georgia Institute of Technology and a B.S. in General Engineering from the University of Illinois at Urbana-Champaign. He is a member of ASEE, IIE, INFORMS and NSPE. He currently serves as an area editor for *The Engineering Economist*.

Lou Plebani

You are currently working on the following problem:

You have been placed in charge of a product line. Your vendor is fairly reliable and generally gets orders to you in 14 days time (+ / - 10%) after the order has been placed.According to a recently signed contract, they will be your sole supplier of product over the next year. Your agreed contract price is 11.78 per unit with a fixed cost of 42.91 per order.

Your task is to determine an ordering and inventory policy. Specifically, determine an order quantity Q and a reorder point R. Your company has a barcode system in place so your order is automatically placed when R is reached. As the lead time of your supplier is not always perfect, it is possible that more than one order may be outstanding at any given time. That is, you may place another order before the previous order(s) has arrived. For this reason, the inventory system will use **inventory position (i.e., current inventory plus all outstanding orders)** for the comparison to the reorder point R.

Your goal is to minimize the sum of all costs, namely sampling cost, the fixed cost of ordering, holding, backlog and all penalty costs. **Please note: your sampling costs are included.** Holding costs are 1.47 per unit per year. If customers wait for their product, a backlog cost of 10.84 per unit per year is paid. However, not all customers will wait for your product, but rather, go to your competition. For each of these customers, you pay a loss of goodwill penalty cost of 22.61 for each sale lost. Historical data shows that out of 100 customers that request a product when there is no inventory, 73 will wait and 27 will go to the competition. Your end of year bonus is tied to how you save money over the next year.

As expected, demand is uncertain. However, for a processing cost of 148.13 per request and a per-unit sample cost of 14.81, the marketing department will supply you with historical data (daily demand). You may request the number of daily observations below. Your may request data more than once but will pay a fixed cost each time. Upon completion of your analysis, enter your values of Q and R.

Request Demand History

Num of Obs [ ]

[Submit Request]

[Review Data Already Obtained]

Enter your solution for this problem and submit

Order Quantity (Q) [ ]

Reorder Point (R) [ ]

[Submit Your Solution for Evaluation]

Figure 1. Problem Story and Action Form

All observations for current problem

197	206	185	195
180	160	217	174
218	173	172	211
209	226	166	198
203	171	186	231

Total number of obs = 20
Average of obs = 193.90
Std. Dev. of obs = 21.02
Maximum of obs = 231
Minimum of obs = 160

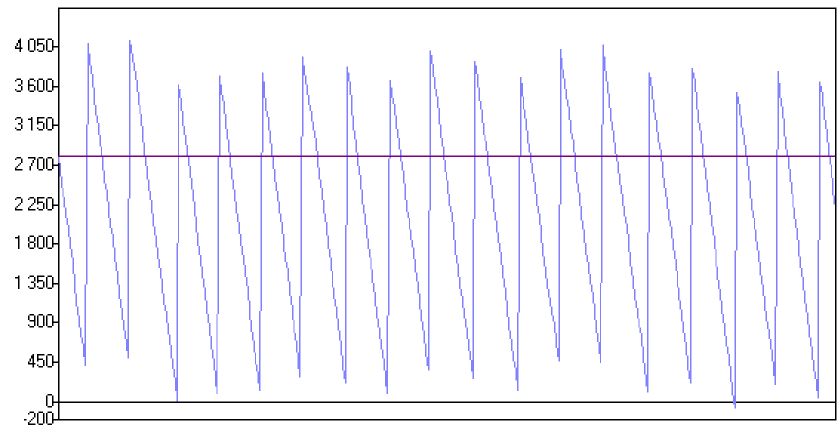
Return to Summary Page

Figure 2. Data Summary Page

Order Quantity = 3800   Reorder Point = 2800   Unit Cost = 11.78   Sampling Cost = 1777.62

Fixed Cost = 42.91   Holding Cost = 1.47   Penalty Cost(/Unit) = 22.61   Penalty Cost(/Time) = 10.84

Stock History



Number of Orders = 19   Sales Lost = 27   Total Order Cost = 772.4

Total Holding Cost = 2973.51   Total Penalty Cost/Unit = 610.57   Total Penalty Cost/Time = 2.26

Your Grand Total Cost = 6136

I solved this same problem using a (Q,R) policy of (2517, 3018). My total cost was **3503**. Therefore, I will give you a score of

(mine/yours)*100 = **57.1**

Continue

Figure 3. Evaluation of Submitted Solution

Please answer each of the following questions then submit

1. Provide your overall impression of the system. Did you enjoy using it? Did it challenge you? Did the problems seem somewhat realistic?

2. Would you be interested in using a system like this to augment your learning experience in this or other courses? Why or why not?

3. Provide any other suggestions, complaints or comments that you feel would help improve the system for future use by students.

Submit Survey

Figure 4. Student Survey