

Performance of laboratory experiments over the Internet: towards an Intelligent Tutoring System on Automatic Control.

Kleanthis Prekas, Maria Rangoussi, Savvas Vassiliadis, and George Prekas

Abstract—Intelligent tutoring systems constitute an evolution of computer-aided educational software. We present here the modules of an intelligent tutoring system for Automatic Control, developed in our department. Through the software application developed, students can perform complete automatic control laboratory experiments, either over the departmental local area network or over the Internet. Monitoring of access to the system (local as well as international), along with student performance statistics, has yielded strongly encouraging results (as of fall 2004), despite the advanced technical content of the presented paradigm, thus showing the potential of the system developed for education and for training.

Keywords—Automatic control, tutoring system, Internet access, laboratory experiments.

I. INTRODUCTION

INTELLIGENT tutoring systems (ITS) have evolved from what are more conventionally known as computer-aided education systems. An ITS exploits artificial intelligence to provide education tailored to the needs or weaknesses of the individual student, [1]. A personalized student model is built on the basis of his/her progress on the course material, and the system adaptively places the emphasis on the student's weak points, offering more instruction or examples, tests, etc. In that sense, an ITS is more interactive, adaptive and responsive to specific student profiles, than conventional computer-aided systems.

The architecture of an ITS typically includes (i) an *expert model*, incorporating the course content in an appropriate knowledge structure, (ii) a *course manager*, responsible for tailoring the course to individual student, (iii) a *student model*, which maintains an updated profile of the student's strong and weak points, and (optionally) (iv) an *instruction environment* for help and instruction in difficulties, [1], [2].

In this paper we present the development of the building blocks of an ITS in the Department of Electronics of the Technological Education Institute of Piraeus / Greece. The focus of our work is currently on the expert model and course manager modules, [4]. The laboratory of the Automatic

Control course (a fourth year undergraduate course) has been selected as the pilot course content, because of certain technical advantages it offers within the department.

The innovation of our approach lies in the applied and practical nature of the chosen content to be taught, namely, the remote performance of the laboratory experiments of an Automatic Control course. Typically computer-aided instruction, as well as ITS systems, has been built around introductory and theoretic instruction material, rather than actual laboratory experimentation, [5], [6]. This approach has allowed developers to focus on (i) the artificial intelligence aspects of the system under development, and (ii) the educational aspects of the procedure. However, as the methods and technologies employed in ITS development mature, moving to instruction of technically demanding material constitutes an interesting challenge. The access and performance statistics, collected over a six-month period during which the system was made available over the Internet, show that our pilot development draws the interest of an audience wider than the students of the department.

II. AUTOMATIC CONTROL LAB EXPERIMENTS

A. Remote Experiment Performance and Simulation

Remote performance of laboratory experiments is made possible thanks to analog-to-digital conversion cards and interface cards, that provide bi-directional low level communication between the user, interacting with a user-level application software, and the actual devices he/she intends to operate during the experiment. In the user – to – device direction, user commands are issued to the software interface, passed on to the interface cards and translated to analog signals that trigger the target device via *actuators*. In the reverse direction, *sensors* capture signals of interest, produced by an operating device; interface and conversion cards translate these signals to digital information and pass it on to the software, in order to inform the user on the experiment progress or results.

Once this "hands-free" operation setup is made technically possible, the second step is to detach the user site from the experimental equipment site. Standard software development procedures, such as the client-server architecture or the three-tier architecture, are exploited to transform the application

K. Prekas, M. Rangoussi and S. Vassiliadis are with the Department of Electronics, Technological Education Institute of Piraeus, Athens GR-12244, Greece, (e-mail: prekas@teipir.gr, tel. +30 201 5381112).

G. Prekas is a student in the Department of Electrical and Computer Engineering of the National Technical University of Athens, Athens, Greece.

software appropriately, so as to execute on separate sites.

The third step of the procedure involves connection of the user site and the device site over a communication network (either a Local Area Network or the Internet). Again, standard computer network technologies are employed to proceed from the second to the third step, [3].

The remote operation described above, however, would not be recommended for untrained users, as it would not meet basic lab safety requirements. Besides safety, cost and availability of certain devices or materials place constraints on their remote use. A viable and educationally advantageous alternative is to *simulate* the experimental setup to the desired degree of detail. The simulated environment can then be used either (i) for training prospective users of the actual setup, or (ii) as a substitute of the actual experiment for educational purposes, so as to avoid heavy use and damage risk of critical equipment. Powerful and efficient simulation tools are now available, allowing for successful simulation of practically any technical system.

In the tutoring system under development, *both* remote experiment *performance* and remote experiment *simulation* routes are pursued in parallel. Our aim is to use remote simulation as a training step for students before proceeding to remote performance. It is worth mentioning here that, in an engineering context, simulation is not able to provide the student with the experience and expertise offered by the actual instruments or devices. Therefore, neither conventional nor intelligent instruction will provide quality education or training, if based solely on simulation.

B. A pilot experiment: Liquid level control.

The Automatic Control lab experiment described below is selected because its simple and clear structure provides a suitable educational example, [7]. The system under control is a liquid tank equipped with a filling valve and a draining pipe with a valve. This is an elementary hydraulic 1st order system. Indeed, the level of the liquid in the tank at time *t*, *h(t)*, is connected to the liquid filling rate, *q_{in}* (in litres/min), and the liquid draining rate, *q_{out}* (in litres/min), through the 1st order differential equation

$$q_{in}(t) - q_{out}(t) = A \cdot \frac{dh(t)}{dt} \tag{1}$$

where *A* denotes the rectangular tank surface.

The system transfer function *G(s)* (Laplace domain) takes the form

$$G(s) = \frac{H(s)}{Q_{in}(s)} = \frac{R}{1 + sAR} \tag{2}$$

where *H(s)* denotes the liquid level, considered as the system output, *Q_{in}(s)* denotes the liquid filling rate, considered as the system input, (both expressed in the Laplace domain) and *R* denotes the draining pipe resistance. It is a linear, time invariant system, with time constant *AR*.

Although the system itself is hydraulic, the liquid level is controlled *electronically*, using potentiometers and voltage / power amplifiers. To control the level of the liquid in the tank, *h(t)*, the open loop system is closed by a feedback loop

(schematic structure in Fig. (1), block diagram in Fig. (2)).

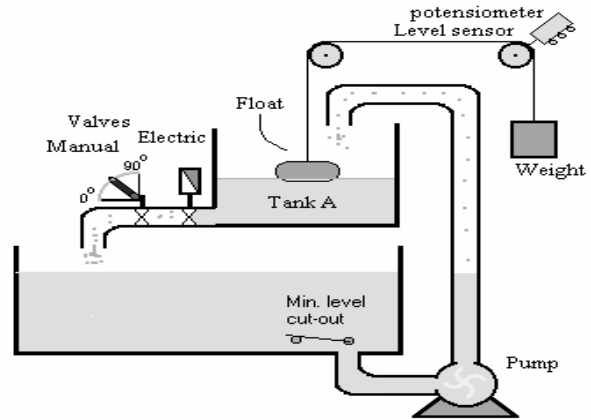


Fig. 1: Liquid level control system schematic (closed loop).

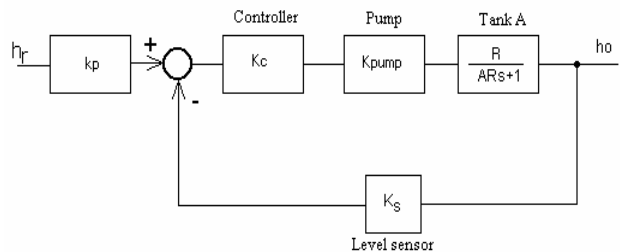


Fig.2: Liquid level control system block diagram (closed loop)

Actual liquid level, *h(t)*, (the closed loop system output) should ideally follow as close as possible the reference liquid level, *h_r(t)*, provided by the user (the closed loop system input). In the closed loop system, a float is used as the level sensor, to monitor *h(t)*, and an electric liquid supply pump along with an electric draining valve are used to control *h(t)*. Drained liquid is reused: the draining valve sends liquid through a draining pipe to a reservoir from which draws the liquid pump.

C. The steps of the experiment

Performance of the experiment proceeds in a sequences of steps, each focusing on a system block or subset of blocks. The student interconnects specific blocks and takes measurements at appropriate points, to identify the block transfer function, time response, or other useful characteristic.

When operating the equipment “hands free” via the software and interface cards, the student is presented with a detailed block diagram view of the system blocks (Fig. (3)). The graphics interface of the software allows the student to *select* and *connect* blocks graphically as well as to connect measurement equipment like voltmeters at selected points, to measure or view results. Equipment not connected is shaded in the background, for clarity purposes.

The steps of the experiment along with the interconnections required for each step are shown in Table I (points are numbered as in Fig. (3)). The first six steps focus on

individual blocks whereas the seventh step refers to measurements over the closed loop system.

TABLE I
EXPERIMENT STEPS & ASSOCIATED INTERCONNECTIONS

| Step No. | Step description | Inter-connections |
|----------|--|---|
| 1 | Preliminary step: Measurement of the pump motor starting voltage | 10-to-9 |
| 2 | Pump interface block: Measurement of the pump interface output as a function of the reference level voltage [$V_{\text{pump-interface}} = f(V_{\text{reference}})$]. | 10-to-7 8-to-M3 |
| 3 | Loop gain amplifier block: measurements for the calculation of the block gain. | 10-to-5 6-to-M3 |
| 4 | Pump and pump interface as one block: measurements for the calculation of the block gain. | 10-to-7 8-to-9 |
| 5 | Liquid level sensor (float) block: measurements for the calculation of the block gain. | 10-to-7 8-to-9 1-to-M3 |
| 6 | Tank and draining pipe as one block: measurements for the calculation of the block time constant. | 1-to-11 |
| 7 | Closed loop system: Recording of the system dynamic response, $h(t)$. Study of the effect of the loop amplifier gain K_c on the liquid level error. | 10-to-3, 1-to-2, 4-to-5, 6-to-7, 8-to-9, 1-to-11, 6-to-M3 |

When the student selects experiment simulation rather than experiment performance, the whole experimental setup and steps remain conceptually the same. The system operation is simulated in the Matlab / Simulink software environment. Student views measured values in simulated voltmeters and the recording of dynamic curves produces plots shown on the screen.

Furthermore, no actual difference can be seen when the student either performs or simulates the experiment over the departmental LAN or over the Internet. After connecting to the Automatic Control Lab website, the student follows the instructions given regarding certain parameter settings and then starts "running" the experiment through the common graphics interface mentioned earlier.

III. INDUCING INTELLIGENCE INTO THE SYSTEM

Remote performance or simulation of laboratory experiments, such as the example described in Section II, is made possible only after having successfully addressed certain technical problems. Predominant among them is the security aspects of remote operation of equipment.

As soon as an acceptable level of secure operation is established, educational quality arises as the next target. Towards this aim we have focused on

(i) Enrichment of the technical content for each lab experiment offered. This includes a wider set of experimental steps, side exercises, (self-) evaluation tests, etc., referring both to parts of the course content and to the whole of it as well. The educational material developed is stored in a database from which it can be automatically extracted upon demand from the application software, based on standard search and retrieval techniques. An elementary expert model module is thus being built on a per experiment basis. Further

improvement includes unification of these per experiment databases, in order to fully exploit technical knowledge common across experiments.

(ii) Provision of an individualized tutoring based on the student profile. This requires a mechanism for building and for updating student profiles along with a mechanism for choosing from within the expert model database the sequence of tutoring material, tests, etc, most appropriate for a specific student profile. We have adopted a rather simplified grading mechanism instead of a complete profiler, [4], whereas for the sequencing of the material we have developed an elementary course manager. Although this solution lacks the full functionalities of an ITS, [5], it offers certain critical advantages: It is "lightweight" and flexible, while it allows for a quick pilot application, in order to collect access and performance statistics, in view of finalizing our ITS design for the Automatic Control Lab.

A. The expert model module

The Automatic Control Lab is an illustrating example of building an ITS, because – among other reasons – it deals with composite *systems* as opposed to elements or devices. The student is required, therefore, to master a variety of background material. For the sample experiment described in Section II (Liquid level control), this material can be roughly organized in three major areas:

- 1) Differential equations basics (math),
- 2) Linear system theory basics, (low order systems, transfer function),
- 3) Electronics basics (amplifiers, gain, measurement equipment),
- 4) Automatic control basics (feedback, closed loop system transfer function).

These areas are common to almost all the experiments of the Automatic Control Lab. An analogous background material organization can clearly be adopted for any other subject, regardless of its specific educational content. The proposed system design includes a basis of auxiliary material for each of the above background areas, including:

- (i) A brief tutorial to the subject, including solved examples,
- (ii) An application example with physical meaning (real world application with realistic values or types of elements / devices / equipment etc.),
- (iii) A low level and two advanced level self-evaluation tests,
- (iv) References for further reading on the subject, ranging from elementary to advanced level.

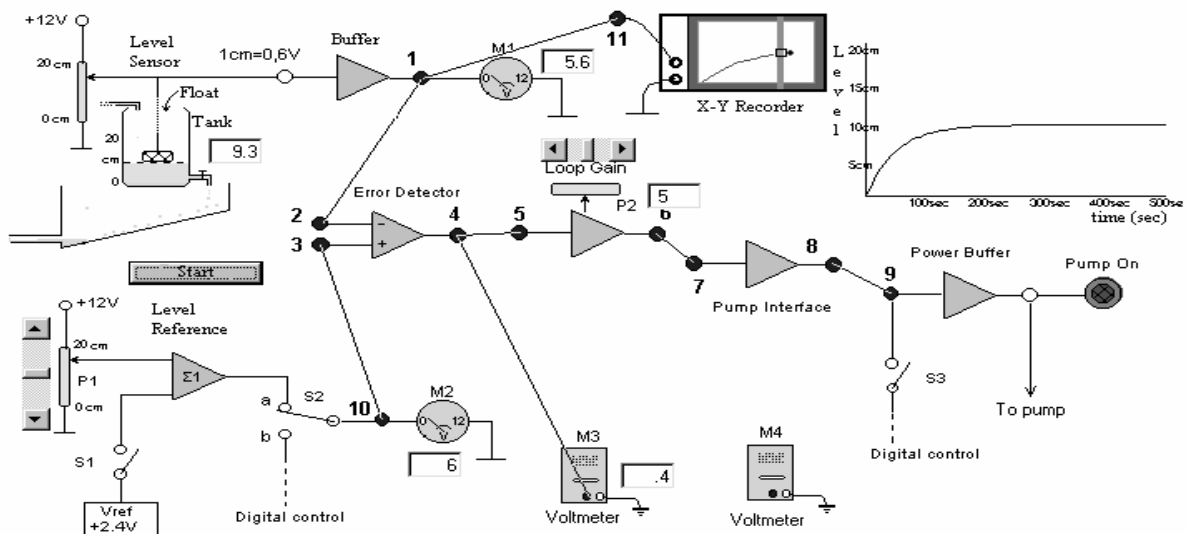


Fig. 3: Screen view of the closed loop system showing all possible interconnections.

B. The course manager module

The student is first presented with one of the advanced level evaluation tests for each one of the background areas. If the student scores above average (the threshold is set to 70%), then the system proceeds to the next background area; if not, then the student is presented with the following sequence of material from the expert model module basis:

- (i) The tutorial,
- (ii) The real world usage example,
- (iii) The low level evaluation test, and (upon success),
- (iv) The second advanced level test.

In case of failure (score under 70%) either at step (iii) or at step (iv) above, the student is directed to the reference material for further reading on the specific background subject. Different references are used in the cases of failure at step (iii) or of success at step (iii) but failure at step (iv). Access to the actual experiment (either for performance or for simulation) is conditional upon success in at least one of the advanced level tests in each one of the background areas.

As mentioned earlier, the design of the system does not currently include a complete student profiler; rather, the crude grading system described above is used for automatic sequencing of the educational material for the specific student.

IV. CONCLUSION

The instruction system presented here has been made available for open experimentation over the departmental LAN and the Internet during the first six months in 2004. It has offered our students a unique experience of computer aided instruction and laboratory experimentation, and as such it has been continuously receiving their strong interest. The access statistics are 21.02% / 65.83% / 13.14% for LAN / international non-USA / USA access, respectively. In absolute numbers, approximately 550 unique visitors have visited the

experimental web site; out of them, around 30 unique visitors per month (average value across the six months) have successfully reached the point of remote experiment performance. These strongly encouraging results exhibit the potential of ITS for technical instruction, including laboratory experimentation, either local or remote. Along this line, we are currently working on the improvement of the artificial intelligence aspects of the system, such as the student profiler and the course manager, aiming to induce to the system a higher level of responsiveness to the individual student.

REFERENCES

- [1] H. L. Burns, and C. G. Capps, "Foundations of Intelligent Tutoring Systems: An Introduction," in *Foundations of Intelligent Tutoring Systems*. Hillsdale, NJ: Lawrence Erlbaum & Associates, 1988.
- [2] P. L. Brusilovsky, "A framework for intelligent knowledge sequencing and task sequencing," in *Proc. 2nd Int. Conf. Intelligent Tutoring Systems (ITS'92)*, Springer-Verlag, Berlin, 1992, pp. 499-506.
- [3] P. L. Brusilovsky, L. Pesin, M. Zyryanov, "Towards an adaptive hypermedia component for an intelligent learning environment." In *Human-Computer Interaction, Lecture Notes in Computer Science #753*, L. J. Bass, J. Gornostaev, and C. Unger, Eds. Berlin: Springer-Verlag, 1993, pp. 348-358.
- [4] K. VanLehn, Z. Niu, S. Siler, and A. S. Gertner, "Student modeling from conventional test data: A Bayesian approach without priors." in *Proc. 4th Int. Conf. Intelligent Tutoring Systems (ITS'98)*, San Antonio, TX, 1998.
- [5] C. Conati, and K. VanLehn, "Further Results from the Evaluation of an Intelligent Computer Tutor to Coach Self-Explanation," in *Proc. 5th Int. Conf. Intelligent Tutoring Systems (ITS'2000)*, Montreal, Canada, 2000.
- [6] K. VanLehn et al., "Minimally invasive tutoring of complex physics problem solving," in *Intelligent Tutoring Systems*, Cerri, Gouarderes, Paraguacu, Eds. 2002, pp. 367-376.
- [7] R. C. Dorf, R. H. Bishop, *Modern Control Systems*. Englewood Cliffs, NJ: Prentice-Hall, 2001, ch. 4, 5.