Copyright © IFAC Advances in Control Education,
Gold Coast, Queensland, Australia, 2000

CONCEPTUAL LEARNING OF CONTROL BY JAVA-BASED SIMULATIONS

J. Sánchez, F. Morilla, S. Dormido, J. Aranda, P. Ruipérez

*Dpto. de Informática y Automática, UNED, Avda. Senda del Rey nº 9, 28040 Madrid.
Spain. Phone: 34-91-3987146 Fax: 34-91-3986697,
E-mail: jsanchez@dia.uned.es, fmorilla@dia.uned.es, sdormido@dia.uned.es*

Abstract: The combination of Java applications as graphical user interfaces and the Matlab-Simulink environment as the simulation engine seems to be a good solution for taking a particular control lab to distant students. In this paper, it is presented a new kind of simulation environment for qualitative teaching on concepts of Control Engineering by means of a suitable configuration of experiments by the tutor or teacher. *Copyright © 2000 IFAC*

Keywords: Educational aids , laboratory, teaching, simulation languages, training.

1. INTRODUCTION

Until now, basically the type of tutorship available in the distance education model has been based on the phone call, postal mail or fax. Nowdays, the *new information technologies* represent the alternative for improving teacher-student interaction. Among all this set of new technologies, two tools can point out as the more complex and reliable when trying to emphasize this new focus within the distance education model: *hypermedia systems* as a new way of arranging information, and the *wide area communication networks* as the information support, that is to say, the Internet network (Maly *et al.*, 97).

Despite the fact that both tools are considered sufficient for the construction of support systems for the learning of subjects without a very strong practical component, the education in Control Systems, or other subjects with strong experimental contents, requires a new element. This new element has to allow students to put into practice the knowledge acquired during the study of the subjects. Nothing new is discovered saying that, in traditional teaching, this role is played by the practice laboratory (Kheir *et al.*, 96; Antsaklis *et al.*, 99). Therefore, if this environment is transfered to distance education, the element needed to put into practice the Control Engineering concepts in a complete way is a *virtual control laboratory* (Patton, and Jayanetti, 96; Schmid, 99). And even, being more ambitious, a *virtual control laboratory with telepresence systems*, that is to say, with the extimulating possibility of taking the remote control of the plants placed at the university lab (Shen *et al.*, 99; Gillet *et al.*, 97; Piguet and Gillet, 99).

Nowadays, there are many commercial and university computer tools for an analytical study of control systems (Johansson *et al.*, 98; Wittenmark *et al.*, 98; Garcia and Heck, 99). Their most outstanding characteristic is that of being quantitative tools demanding that their users have a solid education in Control Systems. This paper seeks to cover an existent hole so far: the necessity of a versatile tool that allows to work dynamically and interactively with some control systems in an intuitive, qualitative, and conceptual way.

In order to describe the tool and its possibilities, this paper is based on the next points: (a) the reasons for the construction of such an environment, (b) the presentation of the elements in the environment used for browsing and experimentation, (c) the different types of experiments for the execution of new practical experiences, and (d) the tools used.

2. WHO ARE THE TARGETS OF THIS CONCEPTUAL LABORATORY ?

This virtual laboratory arises from the necessity of building a system for two very different communities: plant operators and university students. The first group is constituted by people with basic mathematical knowledge (at high-school level) who work, day after day, in their companies with true control systems; their interaction with the industrial environment takes usually place through control panels although, in some occasions, they can have direct contact with the plant. University students are the complete opposite: they have acquired an appropriate knowledge of mathematics and control engineering but, except for the particular practical experiences carried out at the university laboratory (inverted pendulum, DC motor, magnetic levitation), they do not know how industrial plants operate and, therefore, they are unable to imagine which their reactions will be in certain situations. For that reason, a qualitative teaching/training system constitutes a necessary tool for teaching both communities how to work and to act in certain situations and industrial plants.

In spite of the fact that the environment fosters individual work, it is important not to forget the essential role that the trainer or teacher figure carries out. Evidently, a student/operator can interact with the plants of our environment, but without an appropriate focus, the user will not be able to take advantage of all its power and versatility. For this reason, the figure of the teacher/instructor plays an essential role: he is the person in charge of adapting and focusing the content of the training environment according to the formative necessities. As it will be seen later, this will be achieved thanks to the fact that the whole system is parameter-based at different levels.

3. WHAT ELEMENTS HAVE BEEN USED FOR ITS CONSTRUCTION

Qualitative aspects have prevailed over quantitative ones in the project, with the result that the simulation environment is completely interactive and dynamic (no batch or off-line simulations). In this way, the qualitative aspects put on immediately of relief, so much graphically as numerically, in answer to the actions of the users. Summing up, the environment consists of two perfectly differentiated parts (see Figure 1):

- A *powered graphical user interface* (GUI) for browsing and practising. Browsing for searching a particular experiment, and practising for interoperating with the typical objects of an Control Systems laboratory. Basically, each one of these building blocks are abstractions outlining certain industrial elements (plants, controllers, pipes, valves, etc.), panels for the tuning of the controllers and other variables of the process, preprogram events, plus a group of signal diagrams showing the evolution of the parameters during the simulation.

- A *mathematical engine* in which the simulation of certain industrial plants is executed. Since the whole environment is conceived for teaching and training, it is possible to carry out a complete configuration of the models by means of parameter files and, in this way, to define the goals that the students have to reach by manipulating the controls the model presents by means of the GUI. Some of the simulated models are: heat-exchanger with basic control, heat-exchanger with cascade control, level control of a tank, and basic and advanced control of a distillation column.

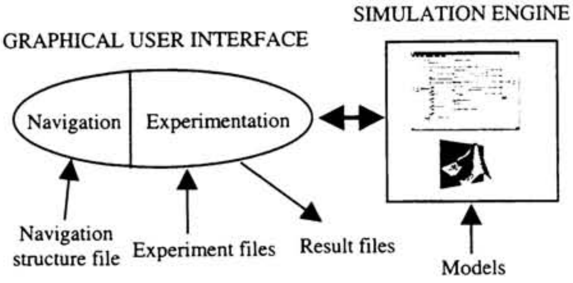
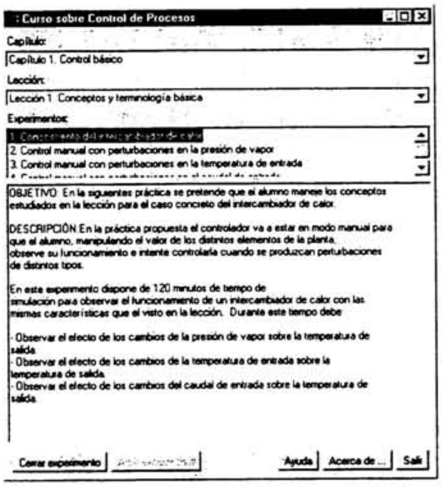


Fig. 1. Outline of the virtual laboratory.

4. BROWSING THE EXPERIMENTATION ENVIRONMENT

To cover the necessities of the two potential users, that is to say, teacher/instructor and student/operator, the teaching/training GUI has two interrelated components, the *browsing* and the *experimentation*, both parameter-based.

The browsing component allows the experiments to be adjusted to the hierarchical structure of a textbook or a course. Three browsing levels are considered: chapters, lessons, and experiments. In this simple way, the teacher/instructor will be able to tailor the environment for teaching a course with a certain profile, and the student/operator will be able to select an experiment among all the existent ones but in an orderly way, according to the concepts or situations that he wants to study or to observe.



Capítulo 1. Control básico

Lección 1. Conceptos y terminología básica

1. Conocimiento del intercambiador de calor
2. Control manual con perturbaciones en la presión de vapor
3. Control manual con perturbaciones en la temperatura de entrada
4. Control manual con perturbaciones en el caudal de entrada

OBJETIVO: En la siguientes práctica se pretende que el alumno maneje los conceptos estudiados en la lección para el caso concreto del intercambiador de calor.

DESCRIPCIÓN: En la práctica propuesta el controlador va a estar en modo manual para que el alumno, manipulando el valor de los distintos elementos de la planta, observe su funcionamiento e intente controlarla cuando se produzcan perturbaciones de distintos tipos.

En este experimento dispone de 120 minutos de tiempo de simulación para observar el funcionamiento de un intercambiador de calor con las mismas características que el visto en la lección. Durante este tiempo debe:

- Observar el efecto de los cambios de la presión de vapor sobre la temperatura de salida.
- Observar el efecto de los cambios de la temperatura de entrada sobre la temperatura de salida.
- Observar el efecto de los cambios del caudal de entrada sobre la temperatura de salida.

Fig. 2. Browsing window.

Figure 2 shows the browsing window of the environment. Its hyperindex-based structure employs menus and allows browsing until selecting a particular experiment. In order to provide more information to the user, simultaneously to the selection of an experiment, a textual description of the objectives of the chosen experiment is provided.

The browsing hierarchy as well as the descriptions of the experiments are located in a plain text file named the *browsing structure file*. Therefore, this file should contain both the browsing hierarchy (chapters, lessons, and experiments) and the definition of the experiments (plants, experiment file, parameter file).

Once the experiment has been selected, the user will begin the practice by means of the graphical experimentation interface. The visual aspect of the interface as well as the process behavior will have been previously established by a teacher/instructor so that the experiment will focus toward the attainment of a certain goal in accordance with the experiment description shown in the browsing window.

5. THE GRAPHICAL EXPERIMENTATION INTERFACE (GEI)

With the purpose of emphasizing dynamism and qualitative aspects, the GEI is characterized by its interactivity with the simulation, the dynamic visualization, the animation of the elements (when the system allows it) and the historical registration of the variables and events. The GEI consists of the following parts: the process diagram, the control panels, the graphical registrations, the panel multisignal and the result file. Next, each one of them is described briefly.

Plant	Interactive variables	Non-interactive variables
Heat-exchanger with basic control	Steam pressure of (bar.) Input temperature (°C) Input flow (m3/h.) Manual control (%) Temperature set-point (°C)	Time of simulation (min.) Flow of input steam (Tm/h.) Output temperature (°C) Opening of the valve (%)
Heat-exchanger with cascade control	Steam pressure of (bar.) Input temperature (°C) Input flow (m3/h.) Vapor flow set-point (Tm/h.) Manual control of TC (%) Manual control of FC (%) Temperature set-point (°C)	Time of simulation (min.) Flow of input steam (Tm/h.) Output temperature (°C) Opening of the valve (%)
Level control of a tank	Input flow (m3/h.) Level set-point (%) Manual control of LC (%)	Time of simulation (min.) Level of the tank (%) Output flow (m^3/h.) Overflowing flow (m^3/h.)
Basic and advanced control of a distillation column	Load flow (m^3/h.) Composition (%C3) Head temp. set-point (°C) Bottom temp. set-point (°C) Reflux manual control (m^3/h.) Steam manual control (m3/h.) L/D manual control V/F manual control	Time of simulation (min.) Head temperature (°C) Bottom temperature (°C) Sludges in the head (%C4) Distilled (m^3/h.) Sludges in the bottom (%C3) Bottom flow (m^3/h.)

Table 1. Process signals of some models.

(a) *Process diagram.* It is composed by a graphical outline of the process (some of the present objects can have animation) with alphanumerical visualization of the most important signals and units, plus an outline of the control strategy with access to the parameters and the mode changes of the controllers. The process diagrams of two plants can be appreciated in Figures 3 and 4, and the most important signals of each one of them are picked up in Table 1. An important aspect of these diagrams is that the schematic representation of the physical elements can change. For example, the tank in Figure 4 can be configured as vertical or horizontal, with the existence or not of a drainage pump, and although it is the same one, its characteristics vary completely.

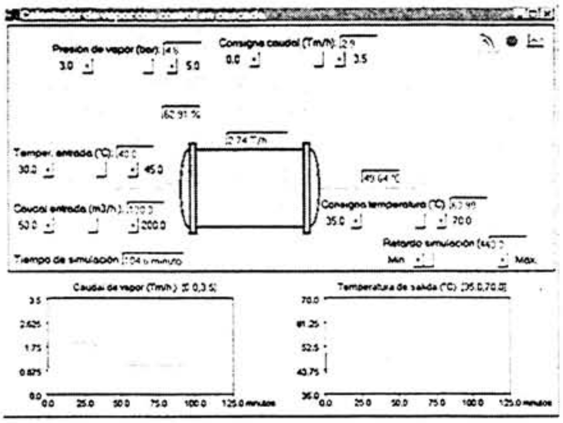

Presión de vapor (bar): 4.5
Consigna caudal (Tm/h): 2.9
Temper. entrada (°C): 40.0
Caudal entrada (m3/h): 100.0
Consigna temperatura (°C): 62.99
Retardo simulación (s): 0
Tiempo de simulación: 104.6 minuto

Fig. 3. Cascade control of a heat-exchanger.

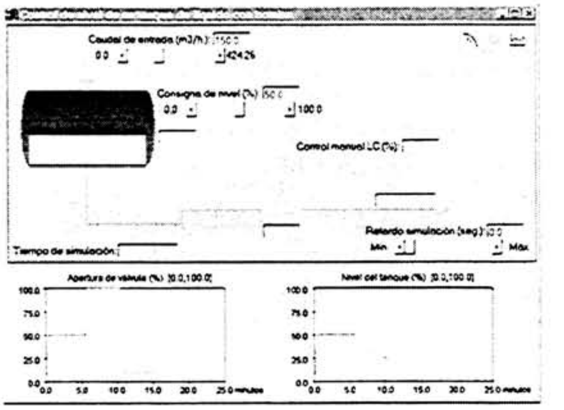

Caudal de entrada (m3/h): 150.0
Consigna de nivel (%): 50.0
Control manual LC (%):
Retardo simulación (seg): 0.0
Tiempo de simulación:

Fig. 4. Interface for level control of the horizontal tank without pump.

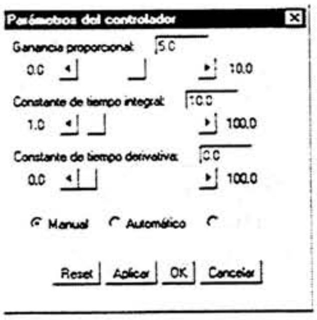
(b) *Control panels.* They have three types of elements (buttons, sliders and fields) that can be grouped in three categories:

- *Control panel.* It allows the stop, continuing and restarting of the simulation process. It is located at the top of the interface.

- *Panel of interactive variables.* It is spread over the whole diagram and consists of sliders and alphanumeric fields to modify the value of the disturbances, the set-points and the manual control signals. With the purpose of allowing the tutor to guide the user's actions, the teacher/instructor can configure the way in which the user interacts with this panel. There are three possible configurations for each interactive variable: totally hidden, visible but not modifiable, visible and modifiable (see the second column in Table 1).

- *Panels of the controllers*. These panels or windows are generally hidden and they are showed when the user clicks on the symbol of a controller (in our case, the controller is represented by a circle with two letters in its inner part indicating the control variable: LC level controller, TC temperature controller, and FC flow controller). As you can see in Figure 5, it is possible to modify the parameters of the controllers in an interactive way as well as the operation mode (manual, automatic or cascade). As it happens with the interactive variables, and in accordance with the aim of the experiment, the teacher/instructor can modify the way in which the user interacts with the controllers: ranges, initial values of the parameters, initial mode of control, direct or inverse control action, etc.

(c) *Graphical registrations*. They provide the graphical visualization of the main variables of the system. At the same time as the simulation advances, these registrations reflect dynamically and continuously any change in the variables that take part in the process. The changes in the system signals can be due to the user's actions (movement of a slider in the process diagram) or to disturbances preprogrammed by the teacher/instructor for this experiment. At the bottom of Figures 3 and 4, the temporal evolution of two signals of the heat-exchanger can be appreciated.



Parámetros del controlador

Ganancia proporcional: 5.0

0.0 ◂ ▸ 10.0

Constante de tiempo integral: 10.0

1.0 ◂ ▸ 100.0

Constante de tiempo derivativa: 0.0

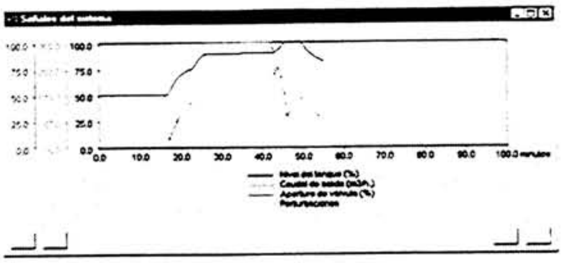
0.0 ◂ ▸ 100.0

● Manual ○ Automático ○

Fig. 5. Controller panel.

(d) *Multisignal panel*. With the intention of giving the user an overall view, the graphical interface has been endowed with a special panel that allows all the signals and disturbances to be showed in an only window (see Figure 6).

An important aspect of all these graphical registrations is that any event preprogrammed by the teacher/instructor is presented in form of a vertical line in the window. In this way, the user can know how and when a disturbance has been introduced in the system and how this has evolved starting from that instant.



Señales del sistema

Fig 6. Multisignal register panel.

(e) *Historical register of events and actions*. Later on, the user or the teacher/instructor will be able to use an analysis tool to study in a quantitative way all that happened during the simulation process.

6. SOME PROPOSAL OF EXPERIMENTS

Once the main attributes of the experimentation interface (dynamism, interactivity and configuration) have been described, it is needed to enumerate some of the exercises that the teacher/instructor can propose to the students with the purpose of studying qualitatively the dynamic characteristics of any of the plants. These exercises can be grouped into three generic categories: (a) knowledge of the process, (b) manual control, and (c) automatic control. Let us detail each one of these points.

a) *Knowledge of the process*. The objective of this kind of exercises is to familiarize the user with the reactions of the process making changes in the disturbances and control signals. Some possible exercises of this type are:

- *Qualitative description*. The student is requested to complete a qualitative matrix of stationary states about the model that he is going to practice. To do this type of exercise, the teacher/instructor will have configured the experiment in manual control, he will have allowed all the interactive variables to be freely manipulated without any type of restrictions, and he will have disabled the controller panels.
- *What input signal has changed?* The teacher/instructor programs a sudden change in an input signal (preprogrammed event), and this variable will remain hidden during the experiment. The user is warned that he has *x* simulation time to observe the output of the process and to discover what input has changed and in what sense (increasing or decreasing). To configure this exercise, the teacher/instructor will have disabled the panels of the interactive variables and the panels of the controllers.

b) *Manual control*. The student will practice with the plant using the controller in manual mode. Previously, the teacher/instructor will have disabled in the controller the possibility of changing the control mode. Some possible exercises are:

- *Control yourself.* The teacher/instructor programs a disturbance in an input variable and the student is warned that he has x minutes to counteract it by means of the opening or closing of the valves. As in the previous case, the disturbed variable will be hidden to the user and he will only be able to see the effect caused in the system. Observe that preprogramming the modification of other signals of the process, varying the number of changes and their instant, and the number of disturbed signals simultaneously, a considerable battery of exercises is obtained.
- *What control action should I use?* The user is requested to analyze the behavior of the process to determine what action type (direct or inverse) should correspond to the controller.

c) *Automatic control.* In this case the teacher/instructor preprograms the experiments so that the user cannot put the controller in manual mode to tune the PID parameters if the behavior of the process is not the appropriate one. The student is forced to fix. Some exercises of this type are:

- *Aim of the regulation action.* The experiment is started with the controllers in automatic mode and the user is requested to make individual changes in all the interactive variables (except the set-points) and to observe if the controller is able to counteract these actions with its current parameters. He is also requested to fill a qualitative matrix of the stationary state of the control system.
- *Aim of the servo action.* The experiment is started with the controller in automatic mode and the user is warned that he carries out modifications in the set-point and observes if the controller is able to make the process variables reach the new set-point.
- *Basic actions with the PID controller.* This kind of exercises seeks to center the attention of the student in the control basic actions: proportional, integral, and derivative. Initially, the teacher/instructor will tune the controller in automatic mode and with control P, PI, or PID. The student will make a change in the interactive variables, observing if the controller is able to counteract these disturbances with its current parameters. Subsequently the user will be requested to remove the disturbances in order to return to the initial state, to increase the controller parameters and to repeat the previous experience of introducing disturbances.
- *Tuning the controller parameter.* There are many exercises in this group. In a generic way, two possible alternatives are: (1) the controller is preprogrammed with some not very appropriate parameters and the user is requested to introduce changes in the interactive variables and to observe the behavior of the system; (2) the user is requested to remove the disturbances in order to return to the initial stationary state, and, after it, to tune the controller parameters for improving the behavior of the control system; he must repeat the experience until getting some good control parameters.
- *Analysis of the limitations in the actuator.* The goal in this kind of exercises is the user to understand that certain control objectives in the real world cannot be obtained because of the physical limitations of the process components.
- *Study of the operation range.* This group includes some experiments similar to the previous one. In this case the goal is that students discover the operation range of any plant and the fact that its control system is always limited.

Some of the exercises enumerated suppose that the control is carried out with a PID controller in manual or automatic mode. For example, this happens in the basic control of the level of a tank (Figure 4). The possible exercises on advanced control have not been enumerated here; two PID controllers are needed in these exercises (for example, the heat-exchanger with control in cascade) and other elements (ratio and feed-forward compensation) as it happens in the distillation column.

7. SOFTWARE TOOLS

The software tools can be grouped into two categories, one for each of the two elements that integrate the environment: the GUI and the simulation engine. For the first one, Java has been chosen for the following reasons (Hamilton, 96):

- The *possibility to develop applications and applets*, as well as the simple process of conversion of one into another. This characteristic allows the whole software developed for its stand-alone operation to become easily an applet.
- Its *orientation to Internet and its APIs for distributed computing*. RMI (Remote Method Invocation) and JavaIDL (Java Interface Definition Language) constitutes a guarantee of the degree of implication in the development of distributed applications.
- Its *objects orientation* offers all the benefits characteristic of this paradigm.
- Its *portability* allows Java applets and applications to be run in any platform that has a Java Virtual Machine.
- The possibility *to extend it with other lower level programming languages*. Thanks to JNI (Java Native Interface), it is possible to develop C

libraries for accessing to peripheral devices or for the programming of real time systems.

As a tool for the simulation and math engine Matlab 5.x and Simulink 2.x have been used. The reason for this election is that both tools can be considered as a facto standard in the field of Automatic Control and Simulation. But there are some more reasons:

- The *great variety of operative systems on which you can use Matlab*. Thanks to that, it is possible to port the Simulink model from a system to another directly.
- *Students can acquire a limited version from both tools at a very low price*. All the simulations in the virtual laboratory have been carried out with these versions, maximizing the distribution of the environment among the students.
- Matlab *interfaces with external languages*. It is necessary to highlight the integration with ActiveX, or the employment of Matlab as a numerical engine from other languages, both in platforms Windows and Unix.

Focusing in ActiveX, Matlab supports the protocol Automation so that it can be controlled and control other components that support ActiveX. When Matlab is controlled by another component, denominated the client, it is said to be a server, and when it controls another component, Matlab becomes the client and the other component the server. In our case, Matlab only interests working as server in order to control it by means of the Java code developed for the simulation interface.

Although the number of development tools in Java is considerable, the Visual Microsoft Java++ environment has been chosen. This has been motivated by the possibility of interconnecting in a simple way the Java code with Matlab, thanks to the support of ActiveX that the Microsoft environment makes.

8. FURTHER GOALS

At the moment, the force task is focused in the development of a graphical user interface with two aims: to help the tutor to configure the browsing by means of the hyperindex contents, and to develop new experiments, focusing the taskforce in the design of new experiments. In addition, it is working in the creation of new models of plants (DC motor, magnetic levitator, etc.) in order to have a new battery of exercises focused on the university world. Effort is also being devoted to the porting of this qualitative virtual lab to a distributed version using Linux machines. Thank to that, students will be able to practice through the network with new experiments and models, avoiding the need that Matlab/Simulink run in their own PCs.

REFERENCES

Antsaklis, P., T. Basar, R. DeCarlo, N. Harris, M. Spong and S. Yurkovich (1999). "Report on the NSF/CSS WorkShop on New Directions in Contrl Engineering Education", *IEEE Control Systems Magazine*, **Vol. 19, nº 5**, pp. 53-58.

Garcia, R.C. and B.S. Heck (1999). "Enhancing Classical Controls Education via Interactive GUI Design", *IEEE Control Systems Magazine*, June 1999, pp. 77-82.

Guillet, D., C. Slatzmann, R. Longchamp and D. Bonvin (1997). "Telepresence: An opportunity to develop practical experimentation in automatic control education", In *Proceedings of the European Control Conf.*, Brussels, 1997.

Hamilton, M.A. (1996). "Java and the Shift to Net-centric Computing", *IEEE Computer*, **Vol. 29, nº 8**, pp. 31-39.

Johansson, M., M. Gäfvert and K.J. Amstrom (1998). "Interactive Tools for Education in Automatic Control", *IEEE Control Systems Magazine*, **Vol. 18, nº 3**, pp. 33-40.

Kheir, N.A., K.J. Åmström, D. Auslander, K.C. Cheok, G.F. Franklin, M. Masten and M. Rabins (1996). "Control System Engineering Education", *Automatica*, **Vol. 32, nº 2**, pp. 147-166.

Maly K., H. Abdel-Wahab, C.M. Overstreet, J.C. Wild, A.K. Gupta, A. Youssef, E. Stoica and E.S. Al-Shaer (1997). "Interactive distance learning over Intranets", *IEEE Internet Computing*, **Vol. 1, nº 1**, pp. 60-71.

Naryanan, S., N. Rao, J. Geist, P. Kiran, H.A. Ruff, M. Draper and M.W. Haas (1999). "UMAST: A Web-Based Architecture for Modeling Future Uninhabited Aerial Vehicles", *Simulation*, **Vol. 73, nº 1**, pp 29-39.

Patton, J.B. and P. Jayanetti (1996). "The Making of Multimedia Power Systems Control and Simulation Labware", *IEEE Transactions on Education*, **Vol. 39, nº 3**, pp. 314-319.

Piguet, Y. and D. Gillet (1999), "Java-based Remote Experimentation for Control Algorithms Prototyping", *Proceedings of the American Control Conference*, San Diego, California, 1999, pp. 1465-1469.

Schmid. C. (1999). "A Remote Laboratory Using Virtual Reality on the Web", *Simulation*, **Vol. 71, nº 1**, pp. 13-21.

Shen, H., Z. Xu, B. Dalager, V. Kristiansen, Ø. Strøn. M.S. Shur, T.A. Fjeldly, J. Lü and T. Ytterdal (1999). "Conducting Laboratory Experiments over the Internet", *IEEE Transactions on Education*, **Vol. 42, nº. 3**, pp. 180-185.

Wittenmark, B., H. Haglund and M. Johansson (1998). "Dynamic Pictures and Interactive Learning", *IEEE Control Systems Magazine*, **Vol. 18, nº 3**, pp. 26-32.