

Web-based Learning Resources for Vocational Training on Control and Measurement Systems: The AutoTECH Project

José Sánchez, Héctor Vargas, and Sebastián Dormido

Abstract— This paper presents a short description of the state of development of a set of interactive virtual and remote laboratories developed in the framework of the AutoTECH project. This EU-funded project is aimed at two specific groups of people of the wide control community: automation technicians and vocational students on measurement and control systems. The chosen process to build the remote lab is a didactical heat-flow setup to make experiments on PID control. Regarding the simulators, three have been the selected processes: self-regulated first and second order processes, and a pure integral process. The selection of these processes is due to all of them are well-known and used systems in the process control industry and, at the same time, they are studied in the European vocational training schools.

I. INTRODUCTION

TILL now, automation technicians did acquired their basic understanding and knowledge on automation and control systems in vocational training studies by enrolling in courses with syllabuses covering different points on control engineering. Yet, currently, and at least in Spain, there is a specific two-year vocational course to become a control technician. Generally, this course is entitled “Automatic Control and Regulation Systems” and some of the control-related subjects covered are basic control techniques, sensors, signal conditioning, PLCs, analog/digital electronic circuits, programming, etc.

Nowadays, automation technicians play an important role in the day-to-day operation and maintenance of automation equipment and control systems in all industrial processes. In spite of that, there is a documented need for specialized and ongoing training and practice in addition to basic education. This is also due to the organization of many companies:

- Automation technicians often have less opportunity of attending training courses to update their skills compared with university graduates.
- Even in large companies there is often only a very small group of people with an automation and control background. This means that automation technicians do not have much opportunity of discussing problems related to their profession with in-house personnel.

Manuscript received September 30, 2006. This work was supported by the Spanish Ministry of Research under Grant DPI2004-01804 and by the European Commission through the Leonardo da Vinci programme under contract N/04/B/PP/165.011 AutoTech.

J. Sánchez (corresponding author, jsanchez@dia.uned.es), H. Vargas and S. Dormido are with the Department of Automatic Control and Computer Sciences of the Spanish National University for Distance Education (UNED), in Madrid.

But, also, there is a constant pressure of the industrial companies to get that educational institutions produce automation technicians with sufficient basic knowledge and experience. The reason for this claim is to minimize as much as possible the companies’ expenditure in formal and non-formal training courses. That can be considered another argument to provide learning resources that are relevant for the industry and that can be easily updated and modified to fit them to the requirements that impose the dynamism of modern industrial world.

With the objective of providing and suggesting some solutions to these problems, six European universities are running a pilot project entitled “AutoTECH: Automation Technicians Vocational Training Repository” which is part of the European “Leonardo da Vinci” programme [1]. The European universities involved in the project are:

Spanish National Univ. for Distance Education (UNED).
Polytechnic University of Valencia (UPV).
Norwegian Univ. of Science and Technology (NTNU).
University of Hagen (FUH).
Polytechnic University of Bucharest (UPB).
École Polytechnique Fédérale of Lausanne (EPFL).

The specific aim of the project is to develop and disseminate a set of new and innovative software packages for automation technicians and vocational students in automation and control. These training packages will allow improving the quality of learning of these two groups of people and increasing their motivation to acquire new knowledge related to their professional careers. Each of these packages will consist of traditional learning resources (theory, exercises, quizzes) and web-based (interactive simulators, remote operable labs, games, etc.) As a result of this project, automation technicians and students will acquire knowledge and practical experience in the design, tuning, and troubleshooting of different industrial control systems.

The goal of this paper is to show up some of the web-based learning materials that UNED’s team has developed in the framework of the AutoTECH project (Section 2). So, next pages describe shortly the remote labs (Section 3) and some of the interactive simulators developed (Section 4). Finally, Section 5 contains concluding remarks and considerations about further works.

II. THE CONTRIBUTION OF UNED TO AUTOTECH

As the AutoTECH members are from different European countries, it has been necessary to fix a set of common subjects that all automation technicians and vocational students on automatic control should know regardless the country they are from. Taking into account the syllabuses of several European countries and the teaching experience and research works of the AutoTECH members, five big common subjects have been defined and so five training packages have been developed to cover every one of these big subjects:

Training Package TP1: Introduction to process control.

Training Package TP2: Basic PID control.

Training Package TP3: Control structures (cascade, ratio, gain scheduling)

Training Package TP4: PLCs. Grafset.

Training Package TP5: Fuzzy control. Application to motion control in mobile robots.

being the UNED team in charge of developing the theoretical contents and practical experiences of the package related to basic PID control.

Since the learning materials are focused to undergraduate students and automation technicians with low knowledge of maths, it has been need to take special care in how to use mathematical abstractions (for instance, the Laplace transform) and the extension and depth of the contents (as example, just single loops are used in basic PID control). Regarding these constraints, the package TP2 “Basic PID control” has been divided into three chapters:

1) Introduction to PID control. It describes the foundations of this well-known control technique, the different control actions (P, I, and D), and the control of first and second order systems using PID controllers.

2) Operational aspects of PID control. Several tuning methods of PID controller in time domain are described. Too, it is explained the programming of PID controllers and how to work out the saturation of the integral action (wind-up phenomenon), the bump-less transfer, the filtering of the derivative, anti-aliasing filters, etc. when digital controller and computers are used in control loops.

3) Analog PID controllers. This chapter is focused in the explanation of pneumatic PID controllers. Too, it describes the design of PID and on/off controllers with neutral zone by connecting op-amps, resistors, and capacitors.

The study of these three chapters is done by three self-explaining documents made up of theory, step-by-step examples, exercises, and open questions. As a complement to this documentation, several interactive simulators and two remote labs have been developed. They will be described

along next sections.

Among all the software tools used to program the virtual and remote labs, it is worth to point up the use of three tools:

- Easy Java Simulations or *Ejs* for short (<http://fem.um.es>) to build the graphical user interfaces of the virtual and remote labs. *Ejs* is an open-source tool specifically designed to create interactive dynamic simulations.

- LabView (<http://ni.com>) to program the real-time control loop running at the server side.

- SysQuake (<http://www.calerga.com>). Tool used to develop two interactive simulators focused to the analysis and design of PID controllers in the frequency domain.

Foundations of these two tools are not explained in this document. To get further information about them and different applications in control education, we encourage readers to get references [3], [7]-[12].

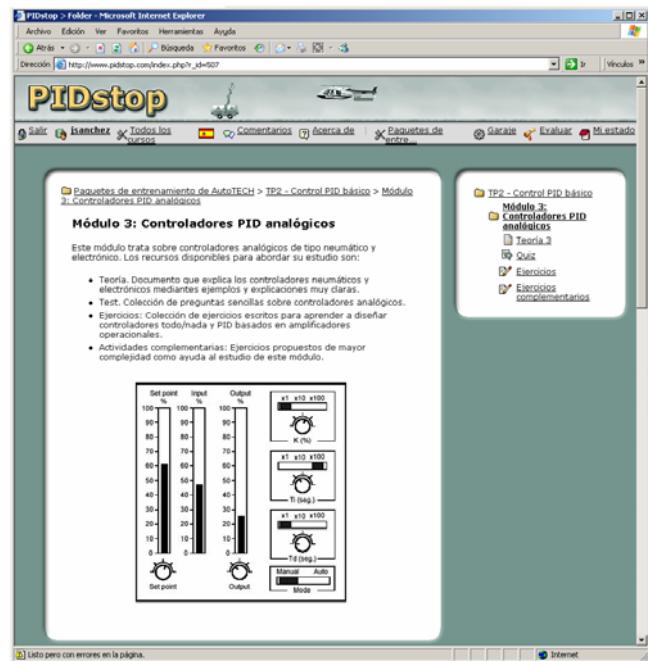


Fig. 1. PIDstop website. The figure shows the introductory page created to give access to all the documentation developed to explain analog PID controllers. On the right side of the page, the different learning resources to study this chapter are enumerated. By clicking on the icons, users will access to the learning resources.

The learning repository and the access to the web-based tools (i.e., virtual and remote labs) are located in a learning portal named PIDstop (<http://www.pidstop.com>) (Figure 1) developed as a prototype at the NTNU. PIDstop can be regarded as a special purpose learning content management system (LCMS) specially designed to fulfil the necessities and functionality needed for automation and control purposes.

III. THE REMOTE LABS

Two remote laboratories have been developed for the learning and training of basic PID control: temperature control of a heat-flow system, and control level using a set of interactive tanks. Both laboratories have been designed to be remotely managed across internet or to be played as simulations using models of both real systems. The reason to switch between virtual and real experimentation in the same user interface is to set up an experimentation protocol that consists of a set of introductory assignments, named *prelabs*. Once passed them, students are allowed to work with the real didactical setups and so to make their experiences using knowledge formerly learnt by working in the simulations with the *prelabs*.

In this section we just describe the remote lab designed for temperature control. Specific information on the remote lab for level control using a three-tank system can be found in [6].

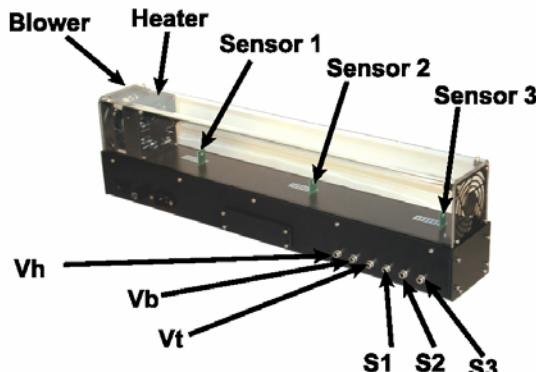


Fig. 2. The heat-flow apparatus by Quanser.

The didactical setup selected to build the virtual and remote lab for temperature control has been the Heat-Flow system by Quanser Consulting (Figure 2). The plant consists of a duct with the following components: a heating element and a blower located at both ends of the structure and three temperature sensors S1, S2, and S3 along the duct. The power delivered to the heater and the fan speed can be controlled using the analog signals Vh and Vb. Fan speed is measured using a tachometer that produces an analog signal Vt and it can be used to design speed controllers.

First of all, it was needed to make identification of the real system in order to get a model of the plant and so build the virtual laboratory. The model obtained corresponds to the following transfer function:

$$G(s) = \frac{C_n(s)}{V_q(s)} = \frac{K_p(1+sT_3)}{(1+sT_1)(1+sT_2)} e^{-Ls} \quad (1)$$

$C_n(s)$: Air temperature at sensor n .

$V_q(s)$: Voltage applied to the heating element.

where steady-state gain K_p , lags T_1 , T_2 , T_3 , and delay L depend on which of the three sensors is used to close the temperature control loop.

A. The client side

Figure 3 shows the main window of the remote laboratory when it is at work in virtual mode that is using the heat-flow model and not the real system. At the top of the window there is a 3D representation of the Quanser heat-flow apparatus whose color varies according to the air temperature inside the duct.

At the bottom of the main window, there is a control panel that allows the user to choose the type of experience (virtual or remote), switch the control mode (open or closed loop), and a set of sliders and buttons to change the operating condition (for instance, changing the set point or introducing some disturbance by varying the heater voltage Vh).

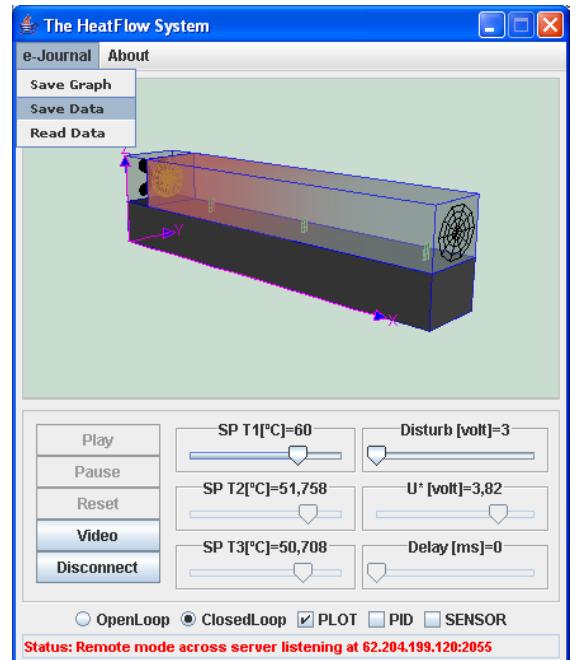


Fig. 3. Graphical user interface (too it is known as the *Ejs* view) developed with the 3D elements of the *Ejs* graphical library. The colour of the apparatus changes according to the temperature of the inner air. It can be considered as a kind of augmented reality technique to reinforce to the user the feeling of physical presence.

The four buttons located on the left side of the control panel (*Play*, *Pause*, *Reset*, and *Connect/Disconnect*) allow users to specify how the system must work. The *Connect/Disconnect* button lets user switch to work with the real heat-flow system running in the university lab or with the simulation running in its computer. Both simulation and remote results are alike because the simulation represents the true behavior of the real system.

The six sliders allow setting the values of the set-point of

each temperature sensor ($SP\ T1$, $SP\ T2$, and $SP\ T3$), the voltage supply to the blower (U^*), the voltage applied to the heater ($Disturb$), and to slow down the simulation ($Delay$). With the two radio buttons, *OpenLoop* and *ClosedLoop*, it is possible to select the operation mode regardless of whether the user is operating the simulation or the real plant. When closed-loop mode is on, the slider U^* is disabled since the controller will calculate the voltage to the blower. Otherwise, the slider U^* will be enabled and the value will be introduced by hand.

Also, there are three check buttons to open the auxiliary windows. By selecting *PLOT*, a window with two scopes is displayed presenting the main variables (temperatures at the sensors, control action U^*); the *SENSOR* button lets users choose a sensor to close the temperature control loop; by selecting *PID*, another window is open with the parameters of the PID controller associated with the selected sensor. Finally, there is a pop-up menu with commands to save the changes in the system variables in a GIF image (*Save graph*) or in a plain text file (*Save data*), and to open an information panel to visualize messages with the state of the connection between client-side (the *Ejs* view) and server side (the didactical setup).

Figure 4 presents two graphical user interfaces of the remote lab when users work with the didactical setup instead of the heat-flow simulation. In this case, the 3D view of the heat-flow is replaced for a real-time video image of the apparatus. Since the air heating process can not be appreciated on the image, too it is possible to overlap the video and the 3D view and so to appreciate the air temperature.



Fig. 4. Two views when the real didactical setup is used for experimentation instead of the model. The view on the right shows how the 3D representation of the heat-flow apparatus is overlaid to the view as an augmented reality technique to improve the user perception.

Figure 5 presents two views of the remote laboratory developed to experiment with a set of three interactive tanks. The design of the views is very similar to the lab for temperature control. In this lab, users have the option to overlap to the video a 2D representation of the tank and compare and contrast the results of the simulation and the real system.

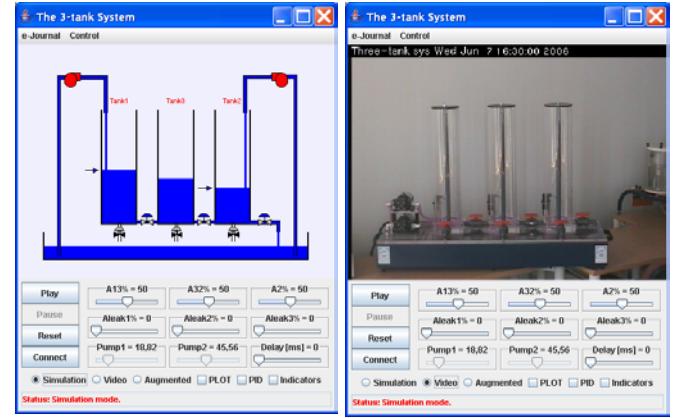


Fig. 5. Two views of the remote lab for level control of a three-tank system.

B. The server side

The design of the server is based on a single client/server architecture, i.e., the same computer acts as a web server and controller (Figure 6). The interface of the client-side is the *Ejs* interactive view (we explain it in the previous subsection) that makes a TCP connection with the server-side when it works as remote lab (the information loop). From the moment that the TCP connection is established, every user action sends a data array to the server, and a vector with the current state of the plant is obtained as a response. By using the parameters of the data packet sent from the view across the information loop, the control loop thus governs the heat-flow system.

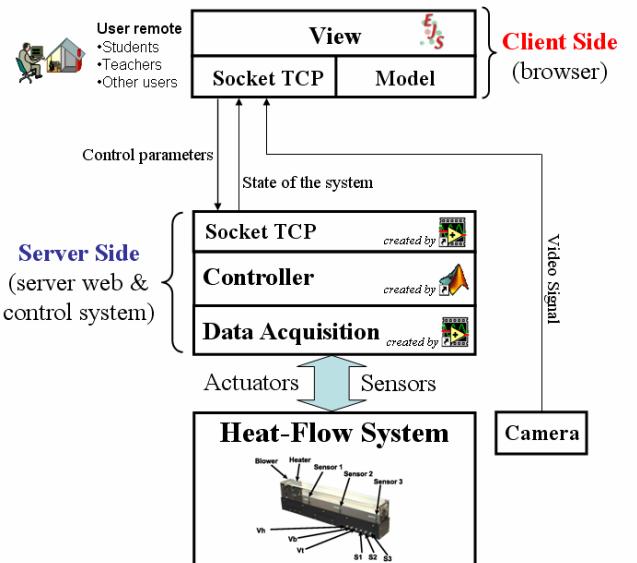


Fig. 6. Outline of the architecture at the server-side. In this scheme, the information loop between client side and server-side (closed across internet) and the control loop (running at the server) can be clearly appreciated and differentiated.

The control parameters sent to the controller are: control mode (manual/automatic), PID parameters (K_p1 , T_i1 , T_d1 , K_p2 , T_i2 , T_d2 , K_p3 , T_i3 , T_d3), blower manual voltage (U_b), and temperature set-points ($SP1$, $SP2$, $SP3$). The information

returned to the view is composed of five values: sampling time (t), temperature measured by sensors (S1, S2, S3), and voltage supplied to the heating element (Uh). The total length of the data packet is 20 bytes.

The control loop at the server-side was developed with LabVIEW and Matlab/Simulink. The data exchange between the computer and the heat-flow apparatus is done by a National Instruments DAQ, and the control loop and the socket connections are programmed by LabView. However, the PID controllers are Matlab/Simulink code. The LabVIEW and Matlab exchange the controller's information using a LabVIEW block to call and evaluate Matlab scripts from LabView programs. This mixed structure is more flexible, and new control strategies can thus be included in the server-side with little programming effort (for instance, to include predictive or fuzzy control in the remote lab, it is enough to get a Matlab/Simulink file with the control code and call it from the LabView block).

IV. THE INTERACTIVE SIMULATORS

Other innovative learning resources developed in the framework of the AutoTech project are the interactive dynamic simulators. The aim of this type of simulators is to offer to students powerful but simple tools to compare and contrast many of the ideas and concepts gathered in the theoretical modules. Since the UNED team is the group on charge of developing the training package to study basic PID control, several interactive simulators of first- and second- order lag processes controlled by PID controllers were created to help students study the theory and complete exercises, quizzes, and open questions.

At the moment of writing this document the simulators fully finished are three ones: two self-regulating first and second order processes and a pure integral process. As physical representation of these processes and so to derive the differential equations representing their behaviour, we have chosen hydraulic models of cylinder-shaped tank, and as control objective we propose to control the amount of liquid stored in the tanks, i.e., the liquid level.

A. Simulator of a self-regulating first order process

The process modeled by the *Ejs* built-in equation editor is an oil tank with an inlet valve to control the input flow plus a smooth tubing acting as an output restriction [2]. Taking into account the values of the default parameters fixed in the simulator, the resulting differential equation that represents the behavior of this process is:

$$113 \frac{dh}{dt} + h = 0.813q_{in} \quad (2)$$

where h is the level liquid in the tank as a percent of the full-scale range of the height of the tank (by default, 2.8 meters), and q_{in} is the input flow rate expressed as percent of the full-scale range of the input flow (by default, 0.4 litres/sec). The

resulting transfer function is:

$$\frac{H(s)}{Q_{in}(s)} = \frac{K_p}{1+Ts} = \frac{0.813}{1+113s} \quad (3)$$

Regarding the automatic control modes used to maintain the level of the tank around a reference, we have programmed three controllers in the simulator: two-position or on/off, proportional, and proportional-integral. The on/off controller includes a neutral zone that can be configured interactively by the user. This feature lets students observe the impact of the hysteresis in the oscillation of the controlled variable (the level) regarding the limits of the differential gap.

As well as we did with the tank, the built-in *Ejs* equation editor has been used to model a continuous PI controller. To obtain it, a state space model of a PI controller has been used:

$$\begin{aligned} \dot{x}_c &= y_{sp} - y \\ u &= \frac{K}{T_i} x_c + Ky_{sp} - Ky \end{aligned} \quad (4)$$

Due to the student's profile that the learning resources are addressed to, the PI and PID controllers do not include concepts such as set-point weighting, bumpless transfer between manual/automatic mode or between controller parameters, filtered derivative action, etc. However, we have decided to include several possibilities to produce the final control action as consequence of its sheer interest to understand how a controller acts at work. So, a controller can be played with or without limitation of the value of the control action, or with anti-windup mechanism if there is integral action.

Also, to include extra features in the simulator, i.e., load disturbances, a second inlet was considered in the model. Too it is possible to change the type of valve used to regulate the input flow (linear, equal percentage, and fast-opening), but just the static characteristics of the valves have been considered, ignoring its dynamics (it would raise the order of the process and only first- and second-order processes are included in the syllabus). We believe the use of nonlinear valves will show up to students the need of using other simple but more robust control techniques (e.g., gain scheduling) to deal with special operating conditions (for instance, non-linearities).

Figure 7 presents the main window of the first-order lag virtual lab. Likewise in the remote lab, this window has two parts: the process diagram and the control panel. The process diagram is dynamic and interactive since it allows users to modify references and disturbances by drag-and-drop techniques and observe the changes in the oil level. The control panel has buttons and auxiliary windows to set up different options of the simulation: buttons to

play/pause/reset, scopes with the most important signals, type of valves, process parameters, simulation options, control modes, and disturbances (Figure 8).

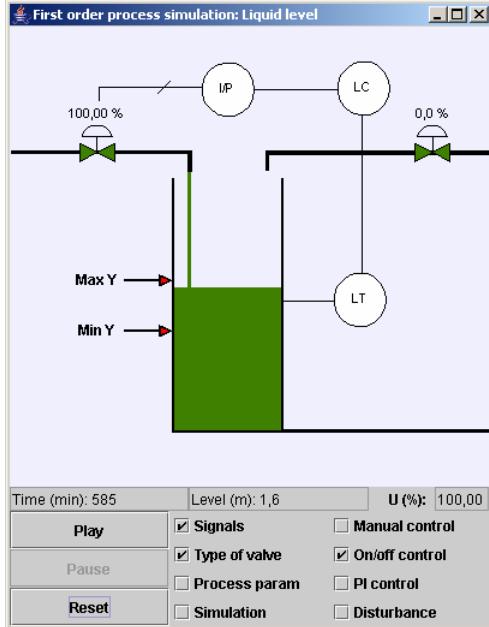


Fig. 7. View of the self-regulating first order process. In this example, the on/off controller is used and the two limits of the neutral zone, both represented by arrows, can be appreciated on the left side of the tank.

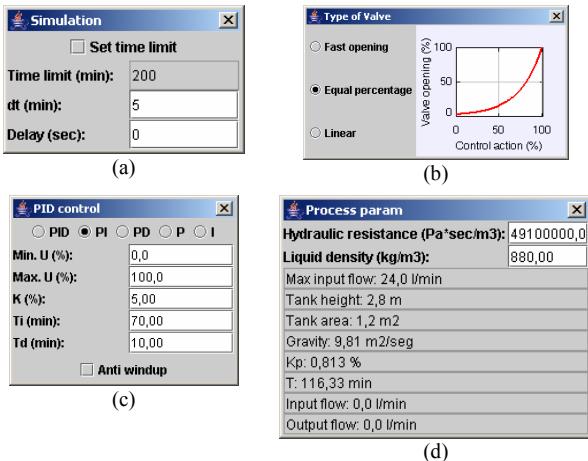


Fig. 8. Auxiliary windows to modify the simulation options (a), the type of valve (b), the controller parameters (c), and the process parameters (d).

B. Simulator of an integral first order process

To model an integral first order process, we have used an oil tank with similar characteristics to the previous one but adding a pump to the outlet in order to maintain a steady output flow regardless the input flow [2].

In relation to the features of the simulator view (Figure 9), the only difference with the previous simulator is the possibility to modify the output flow.

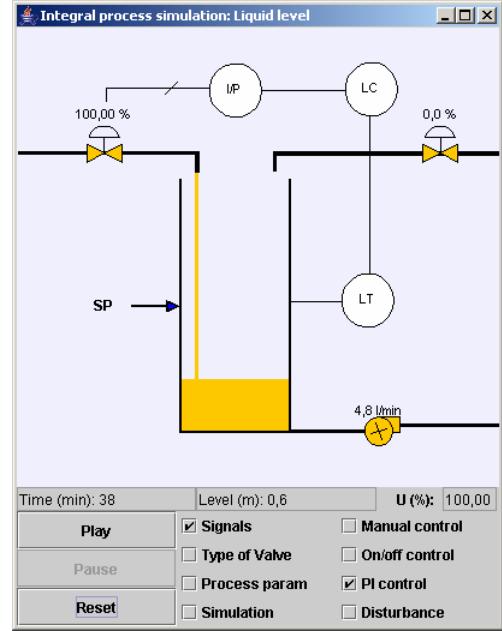


Fig. 9. View of the integral first order process. The only difference with the self-regulating first order process simulator is the possibility of modifying the output flow by adjusting the velocity of the outflow pump (check button "Process param.").

C. Simulator of a self-regulating second order process

The third simulator uses a second order process based on two capacity elements, i.e., a cascade combination of two non-interacting tanks (Figure 10). The functionalities of this simulator are the same that the self-regulating first order but using a PID controller.

In order to include the continuous PID controller with a first order filter in the derivative action, we have programmed the state space model of this controller using the built-in *Ejs* equation editor:

$$\begin{aligned} \dot{x}_1 &= -\frac{1}{T_f} x_1 - \frac{1}{T_f} y \\ \dot{x}_2 &= y_{sp} - y \\ u &= -KN(x_1 - y) + \frac{K}{T_i} x_2 + K(y_{sp} - y) \end{aligned} \quad (5)$$

where $T_f = T_d/N$ is the time constant of the first order filter.

V. CONCLUSIONS AND FURTHER WORKS

Until now, the bigger research and development efforts on web-based learning resources have been oriented to university students and researchers, keeping aside the requirements and expectations of non-university education levels. The development of virtual and remote labs for control engineering education is considered a mature technology. Yet, to date, the application of these learning components to non-university education has not appeared in control education journals or conference proceedings. It has meant that a wide and significant group of the control

engineering community, that is, automation technicians and vocational students, has been ignored. The AutoTECH project is an innovative initiative that pretends to move and apply the knowledge available about web-based control education to this educational level.

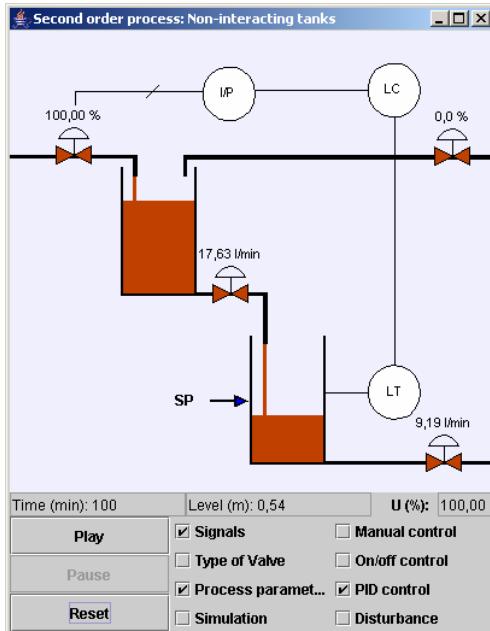


Fig. 10. Self-regulating second order process represented by a cascade combination of two non-interacting tanks.

It is necessary to mention that every AutoTECH's learning resource must be available in English and in the mother tongues of the project participants (Spanish, Norwegian, German, Romanian, and French). Thus labs and simulators described in this paper have been developed taking into account this requirement. When a user gains access to the AutoTECH portal and chooses the language, the remote labs and simulators make use of the *Ejs* internationalization features to change the text labels and messages in the view.

As a complementary resource to the simulators developed by *Ejs*, we are programming two interactive simulators to explain PI and PID control of first and second order systems, respectively, in the frequency domain [4], [5]. The tool used is SysQuake (<http://www.calerga.com>) and we can consider this tool as the perfect one for such a kind of interactive simulators. Figure 11 presents the view of one of these prototypes.

Again, *Ejs* has been shown to be a suitable tool for developing interactive virtual/remote labs. Also, the latest version now includes instrumentation and control symbols in the graphical library. Thus the developer can easily make up virtual and remote labs with views that match the standards of the industrial process control world. A complete English manual on the software tool, Easy Java Simulations, can be downloaded for free from *Ejs*' web server at <http://fem.um.es>.

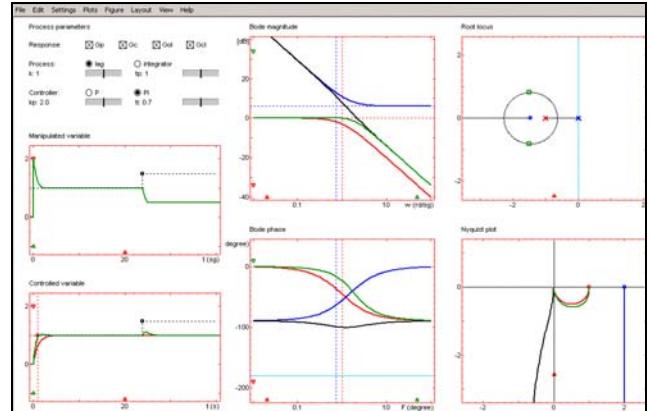


Fig. 11. Simulator for analysis and design of PI controllers with first order systems. The interface is totally interactive since it lets users obtain the PI parameters by dragging on the different representations of the closed loop system (Bode, Nyquist, root locus). Too it is possible to visualize the system response in the time domain and, so compare and contrast both system representations.

REFERENCES

- [1] AutoTech (2006) [Online]. Available: <http://www.pidstop.com>
- [2] R. N. Bateson, *Introduction to Control System Technology*, 7th ed., Prentice Hall, 2001.
- [3] D. Buccieri, J. Sánchez, S. Dormido, P. Mullhaupt, and D. Bonvin, "Interactive 3D Simulation of Flat Systems: The SpiderCrane as a Case Study," in 44th IEEE CDC/ECC, 2005.
- [4] S. Dormido, "Control learning: Present and future", *Annual Reviews in Control*, vol. 28, no. 1, pp. 115-136, 2004.
- [5] S. Dormido, S. Dormido-Canto, R. Dormido, and J. Sánchez, "The Role of Interactivity in Control Learning", *International Journal of Engineering Education*, vol. 21, no. 6, pp. 1122-1133, 2006.
- [6] N. Duro, R. Dormido, H. Vargas, S. Dormido, J. Sánchez, and R. Pastor, "The Three-Tank System: A Remote and Virtual Control Laboratory using Easy Java Simulations", in 44th IEEE CDC/ECC, 2005.
- [7] F. Esquembre, "Easy Java Simulations: A software tool to create scientific simulations in Java", *Comp. Phys. Comm*, vol. 156, pp. 199-204, 2004.
- [8] J.L. Guzmán, M. Berenguel, and S. Dormido, "Interactive teaching of constrained generalized predictive control", *IEEE Control Systems Magazine*, vol. 25, no. 2, pp. 52 - 66, 2005.
- [9] J.L. Guzmán, M. Berenguel, F. Rodríguez, and S. Dormido, "Web-based remote control laboratory using a greenhouse scale model", *Computer Applications in Engineering Education*, vol. 13, no. 2, pp. 111-124, 2005.
- [10] J.L. Guzmán, K.J. Åström, S. Dormido, T. Hägglund, and Y. Piguet, "Interactive learning modules for PID control" in 7th IFAC Symposium on Advances in Control Education, 2006.
- [11] J. Sánchez, S. Dormido, and F. Esquembre, "The Learning of Control Concepts Using Interactive Tools", *Computer Applications in Engineering Education*, vol. 13, no. 1, pp. 84-98, 2004.
- [12] J. Sánchez, F. Esquembre, C. Martín, S. Dormido, S. Dormido-Canto, R.D. Canto, and R. Pastor, "Easy Java Simulations: An Open-Source Tool to Develop Interactive Virtual Laboratories Using Matlab/Simulink", *International Journal of Engineering Education*, vol. 21, no. 5, pp. 798-813, 2005.