

# THE QUADRUPLE-TANK PROCESS: AN INTERACTIVE TOOL FOR CONTROL EDUCATION

S. Dormido\*, F. Esquembre<sup>†</sup>

\*Universidad Nacional de Educación a Distancia. Departamento de Informática y Automática  
Avenida Senda del Rey 9, E 28040, Madrid, Spain  
Phone: +34 91 3987151  
Fax: +34 91 3986697  
E-mail: sdormido@dia.uned.es

<sup>†</sup> Universidad de Murcia. Departamento de Matemáticas  
Campus de Espinardo, E 30071, Murcia, Spain.  
Phone: +34 968 363534  
Fax: +34 968 364812  
E-mail: fem@um.es

**Keywords:** Control education, Interactivity, Web-based simulation, Virtual laboratory, Quadruple-tank process.

## Abstract

This paper deals with the use of interactive computer simulation and animation environments in a virtual lab which illustrates real-world applications of fundamental control principles. The goals of the paper can be summarized as follows: a) Provide students with a motivating introduction to the practice of control engineering principles. b) Show using a simple and challenging example –the quadruple tank– as many features of multivariable control as possible. c) Integrate fundamental skills and principles from throughout the curriculum to solve practical engineering problems. d) Encourage the students promoting an active attitude using problem-based learning via open-ended control problems.

## 1 Introduction

The use of interactive multi-window computer-aids (dynamic picture) in instruction is particularly significant because it provides practical insight into control system fundamentals [7, 8, 18, 21]. They make possible an early introduction of such concepts such as linearization, effects of nonlinearities [6], integrator windup, effects of time delay etc. At the same time, they allow for demonstration of design iterations interactively. For example, a pole on a lead-lag network may be dragged on a root locus plot using the mouse and the effect on the transient response displayed instantly. Hence design iterations using classical methods (root locus, Bode) and modern methods (pole-placement, LQG,  $H_\infty$ ) may be carried out and demonstrated in an instructive fashion. The use of interactive graphical oriented tools would reinforce active participation of students.

Computer simulations in an instructional context mean using the computer to build models or to model real-world phenomena in order to help students gain insights into the behaviour of complex systems. Interacting with an

instructional simulation can enable learners to gain a better understanding of a real system, process or phenomenon through exploring, testing hypotheses, and discovering explanations for the mechanisms and processes. This interactivity may provide opportunities for students to modify their mental models, by comparing the outputs of the model with their expectations, and to engage or motivate students to explore and couple actions with effects which will lead to understanding.

Furthermore, the student has an excellent opportunity to “experiment” with his or her own ideas in terms of engineering design by simple interaction with the tool. Dynamic pictures can not only be effective in presenting engineering concepts in the classroom but also beneficial in extending students experience in analysis and design assignments. This invitation of creativity can be most useful when it comes to specialized control engineering student projects.

In this context, the quadruple-tank introduced by Johansson (Johansson, [14, 15]) has received a great attention because it presents interesting properties in both control education and research. The quadruple-tank exhibits in an elegant and simple way complex dynamics. Such dynamic characteristics include interactions and a transmission zero location that are tunable in operation. With adequate tuning this system presents non minimum-phase behavior that arises due to the multivariable nature of the problem. For this reason the quadruple-tank has been used to show the results of different control strategies and as an educational tool in teaching advanced multivariable control techniques.

The paper is organized as follows. The quadruple-tank process is briefly described in Section 2. Section 3 is devoted to introduce basic concepts of Easy java Simulation (Ejs), a software tool that helps create dynamic, interactive scientific simulation in Java language. The implementation of the quadruple-tank virtual lab in Ejs for interactive control

education is discussed in Section 4. Finally some conclusions are given in Section 5.

## 2 The quadruple-tank process

The quadruple-tank process is a simple connection of two double-tank processes, which are standard processes in many control laboratories (Åström and Lundh, [2]). The setup is thus simple but still the process can illustrate interesting multivariable phenomena. The process flowsheet is displayed in Figure 1.

The target is to control the levels  $y_1$  and  $y_2$  in the lower two tanks with two pumps. The process inputs are  $v_1$  and  $v_2$  (input voltages to the pumps). The model used in the present virtual lab includes also the disturbance effect of flows in and out of the upper-level tanks.

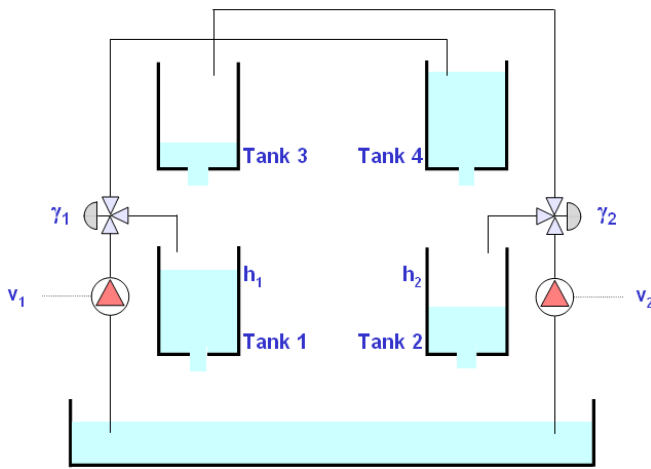


Figure 1: Schematic of the quadruple-tank process

The differential equations representing the mass balances in this quadruple-tank process are:

$$\begin{aligned} \frac{dh_1}{dt} &= -\frac{a_1}{s_1(h_1)}\sqrt{2g|h_1|} + \frac{a_3}{s_1(h_1)}\sqrt{2g|h_3|} + \frac{\gamma_1 k_1 v_1}{s_1(h_1)} \\ \frac{dh_2}{dt} &= -\frac{a_2}{s_2(h_2)}\sqrt{2g|h_2|} + \frac{a_4}{s_2(h_2)}\sqrt{2g|h_4|} + \frac{\gamma_2 k_2 v_2}{s_2(h_2)} \\ \frac{dh_3}{dt} &= -\frac{a_3}{s_3}\sqrt{2g|h_3|} + \frac{(1-\gamma_2)k_2 v_2}{s_3} - \frac{k_{d1} d_1}{s_3} \\ \frac{dh_4}{dt} &= -\frac{a_4}{s_4}\sqrt{2g|h_4|} + \frac{(1-\gamma_1)k_1 v_1}{s_4} - \frac{k_{d2} d_2}{s_4} \end{aligned} \quad (1)$$

where  $h_i$  is the liquid level in tank  $i$ ;  $a_i$  is the outlet cross-sectional area of tank  $i$ ;  $s_i(h_i)$  is the cross-sectional area of tank  $i$ ;  $v_j$  is the speed setting of pump  $j$ , with the corresponding gain  $k_j$ ;  $\gamma_j$  is the portion of the flow that goes into the upper tank from pump  $j$ ; and  $d_1$  and  $d_2$  are flow disturbances from tank 3 and tank 4 respectively, with corresponding gains  $k_{d1}$  and  $k_{d2}$ . The process manipulated inputs are  $v_1$  and  $v_2$  (speed settings to the pumps) and the measured outputs are  $y_1$  and  $y_2$  (voltages from level measurement devices). The measured level signals are

assumed to be proportional to the true level, i.e.,  $y_1 = k_{m1}h_1$  and  $y_2 = k_{m2}h_2$ . The level sensors are calibrated so that  $k_{m1} = k_{m2} = 1$ .

This process exhibits interacting multivariable dynamics because each of the pumps affects both of the outputs. The linearized model of the quadruple-tank process has a multivariable zero, which can be located in either the left or the right half-plane by simply adjusting the throttle valves  $\gamma_1$  and  $\gamma_2$ . Johansson showed that the inverse response (non minimum phase) will occur when  $0 < \gamma_1 + \gamma_2 < 1$  and minimum phase for  $1 < \gamma_1 + \gamma_2 \leq 2$ . The valve settings will give then to the overall system entirely different behavior from a multivariable control viewpoint. Unmeasured disturbances can be applied by pumping water out of the top tanks and into the lower reservoir. This exposes students to disturbances rejection as well as reference tracking.

Using the multivariable quadruple-tank process different aspects of multivariable control systems can be illustrated. For example:

- The influence of the valve settings on the location of the zeros.
- Recognize when a process is easy or not to control [10].
- The influence of the locations of the zeros on the system response for different input directions [15].
- Design and evaluation of decentralized control [15, 22].
- Design and evaluation of decoupling controllers [22].
- Design and evaluation of state feedback controllers for different locations of the zeros [1].
- Design and evaluation of model based predictive control strategy [4, 10, 12].
- Design and evaluation of  $\mu$ -analysis-based  $H_\infty$  control [22].

## 3 Easy-Java simulations fundamentals

*Easy Java Simulations*, Ejs for short, is a software tool that helps create dynamic, interactive scientific simulations in Java language (Esquembre, [11]). The tool is targeted for both science students and teachers with basic programming skills and is therefore very much suited to the pedagogical situation one finds in most university classrooms.

Within Ejs, simulations are created by specifying a model for the simulated system and by building a view that continuously visualizes the state of this model and that readily responds to user interaction.

Most computer simulations of scientific phenomena can be described in terms of the **model-control-view** paradigm. This paradigm states that a simulation is composed of three parts:

1. The **model**, which describes the phenomenon under study in terms of
  - variables, that hold the different possible states of the phenomenon,
  - relationships among these variables (corresponding to the laws that govern the phenomenon), expressed by computer algorithms.

2. The **control**, which defines certain actions that a user can perform on the simulation.
3. The **view**, which shows a graphical representation of the different states that the phenomenon can have. This representation can be done in a realistic or schematic form.
3. Describe how the value of these variables change in time, and
4. Establish how they affect each other when the user interacts with the system and modifies one or more of their values.

These three parts are deeply interconnected. The model obviously affects the view, since a change in the state of the model must be made graphically evident to the user. The control affects the model because control actions can (and usually do) modify the value of variables of the model. Finally, the view affects the model and the control, because the graphical interface can contain components that allow the user to modify variables or perform the predefined actions. In fact, going a step further in the process of simplifying the construction of a simulation, Ejs suppresses the control part, merging it half into the view, half into the model. Actually, modern computer programs are interactive, which means that the user can modify the program's logic by doing some gestures (such as clicking or dragging the mouse, or hitting the keyboard) with the computer peripherals on the program's interface (or view). Thus, the view itself can be used to control the simulation. On the other hand, if we want this interaction to have certain relevance on the program, these gestures on the interface need to trigger actions that affect model's variables. Therefore, the best place to define these actions is in the model itself.

Creating a simulation in Ejs consists in defining its model and its view, establishing the mutual connections needed for

- a) the correct visualization of the state of the phenomenon being simulated and
- b) the appropriate interaction of the user with the view (either to modify this state or to perform the actions defined on the model).

This explicit separation in parts reinforces conceptually the central role of the model of a simulation. It is the model which defines what the program simulates and how. Notice also that there may be different views for a given model. Teacher can create the same simulation with different view for different task or different students. It also makes the task of creating a simulation more modular and promotes reusability, since the parts can be created independently in time, or by different people.

The tool provides extensive scaffolding for creating the model but still leaves full flexibility – and responsibility – to the creator. This is pedagogically important, since the process of learning good control fundamentals consists, to a great extent, in to know the basic principles to build models. In order to describe a model, a user of Ejs needs to be able to

1. Identify the set of variables<sup>1</sup> that properly describe the system
2. Initialize, in a correct way, these variables

---

<sup>1</sup> The word *variables* means here computer variables, which may correspond either to constants, parameters or state variables in physical terminology.

With respect to the view, Ejs offers a high degree of automatism when designing and generating a user interface for the simulation. Even so for sophisticated interfaces, as was the case, since our view had to display the many different possible states of the model but also serve as the interactive front-end for the various interactions that the learning process for this control system demands.

Providing this help is important because students (but also teachers) can then concentrate most of their time in writing and refining the algorithms of the underlying scientific model and dedicate the minimum possible amount of time to the programming techniques (such as multi-threading, handling of interface events, use of different libraries,...). And by doing so, still obtain an independent, high performance, Internet-aware, final product.

The choice of Java as development language is justified in terms of its wide acceptance by the international Internet community, and the fact that it is supported under several software platforms. This implies that simulations created using Ejs can be used as independent programs under different operating systems, or be readily distributed via Internet and run as *applets* within html pages by using any of the most popular web browsers [19,20].

#### 4 Implementation of the quadruple-tank virtual lab using Easy-Java simulation

This section briefly describes the main functions of the developed tool programmed with Ejs. When developing a tool of this kind, one of the most important things that the teacher has to have in mind is the structuring of the main windows and menus of the tool in order to facilitate the student the understanding of the virtual lab (Dormido, [5]). In this case the view, in Ejs terminology, is divided into several parts represented in Figure 2.

Because the simulation is to be used, mainly for didactical purposes, it was very important for us to provide a visualization that displays as realistically as possible the physical system but that also provides support for graphic tools commonly used in control (plots, diagrams, input devices, controllers ...)

The view is therefore structured in one main window and four secondary dialog windows. The main window can be seen in Figure 3. In its upper part we schematically display the four tanks and the system of dripping holes, pipes, pumps and valves that configure the system. We tried to make this first image as self-explanatory as possible. The black arrows point to the set point of the lower tanks, those that have to be controlled. These set points can be changed interactively by dragging the arrow up and down.

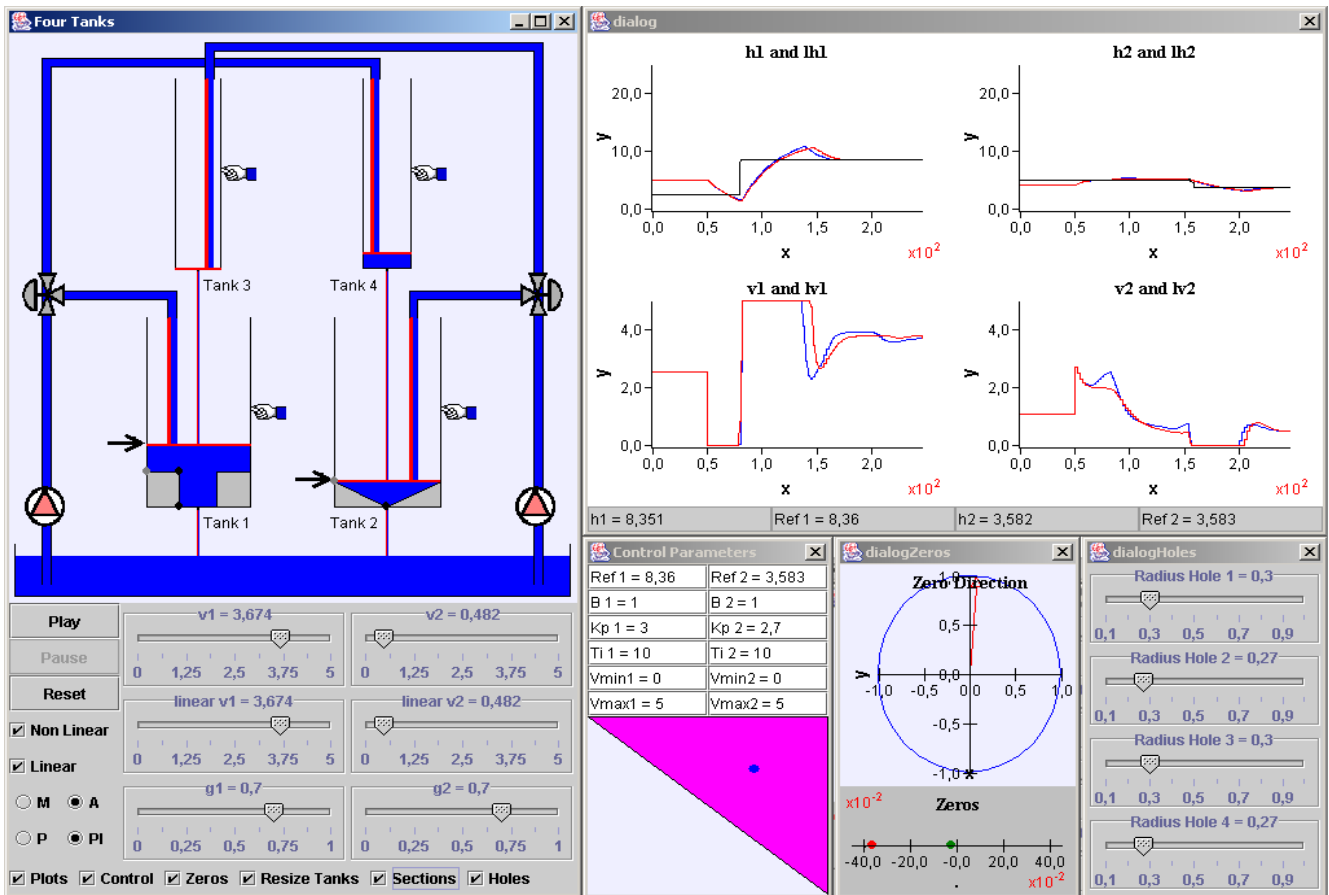


Figure 2: Quadruple tank virtual lab

The lower part of the window is occupied by primary controls that let the user specify how the system should run. With the left-hand side column of buttons the user can make the following actions:

- Play, pause and reset the simulation
- Select to run the non-linear or linear models (or both) for the system
- Choose whether to control the system manual (M) or automatically (A)
- Select, in the case of automatic control, whether to use a P or PI controller (this can be extended to other type of controllers in a very easy way).

If the system is run in manual mode, the user needs to directly set, using the sliders provided in this main window, the voltage for the pumps and the  $g_i$  parameters for the valves ( $g_i$  is the proportion of the flow of pump  $i$  which goes into the corresponding lower tank), so that to reach the prescribed set points. Finally, the lower row of check boxes help bring to view further dialogs and controls. Because it is important in teaching control systems to consider linear versions of our non-linear problems, the view can also display the state of the linear phenomenon.

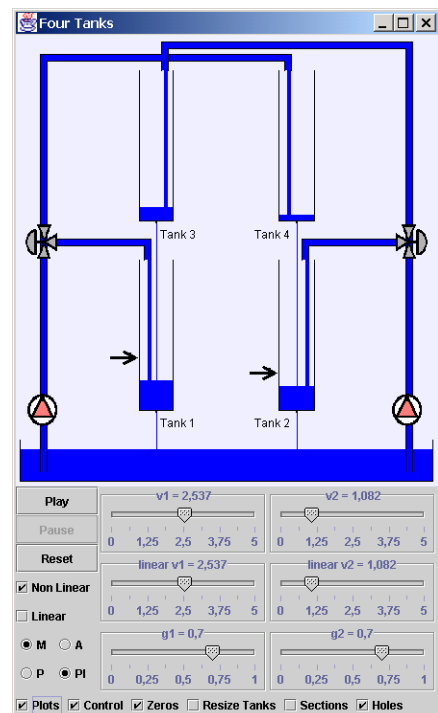


Figure 3: Main window of the view

If asked to do so (by checking the corresponding box) we would see superimposed levels for the linear problem. When set to play, the system solves the equations for the system, both linear and non-linear, for the prescribed values of the control parameters and dynamically displays the level of the liquid in each of the tanks, together with the liquid that gets in and out of them (or none, if this is the case). Blue liquid for the non-linear case and red liquid for the linearized model.

The temporal evolution of the system in time is also visualized in a separate dialog. In it, the levels of the lower two tanks and the voltages applied to the pumps, both in the linear and non-linear case, are plotted against time. See the upper-right window in Figure 2 for a typical run.

**Changing the physical properties of the system.** The main physical parameters of the system can be changed interactively. These are the diameters of the four tanks, their geometric shape, and the diameters of the dripping holes at their bottoms. Clicking the *Resize Tanks, Sections and Holes* boxes bring-in the appropriate controls (see Figure 4).

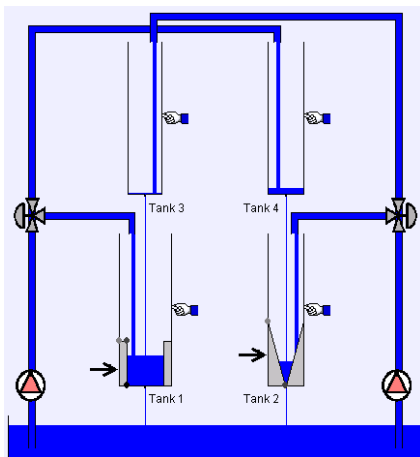


Figure 4: Sections and sizes can be changed dragging the hands and the control circles

The first two consist of graphical gadgets (in the form of a hand and of control circles, respectively) that can be dragged to change the corresponding parameter. In the third case, because their size is much more reduced, a set of sliders help accomplish the task. Figure 4 shows the effect of resizing the first tank as well as two of the possible sections for the lower tanks. This possibility makes the control problem more difficult (the use of gain-scheduling strategy is one of this alternative)

**Controlling the system.** The simulation addresses its main pedagogical target when it is set in automatic mode. Obviously, manually taking the system to desired set points is very hard to achieve, if not impossible, due to the effect of the liquid that simultaneously gets into the upper tanks (and then drip). However, the system behaves very nicely when applying a P or PI control. Further dialogs are devoted to display diagrams and to help adjust the parameters for this automatic controls. In the dialog displayed in Figure 5, the diagram displays the position of the point  $(\gamma_1, \gamma_2)$ .

This diagram is interactive since dragging the blue circle also changes simultaneously the valve parameters  $\gamma_1$  and  $\gamma_2$ . Crossing the diagonal has important consequences: above the system is minimum phase (easy control problem) and below it is nonminimum phase (difficult control problem).

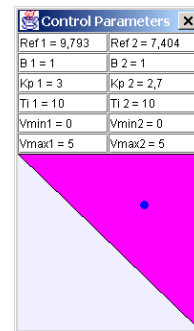


Figure 5: Controller parameters and setting of the valve parameters

A great difference between SISO and MIMO systems is that not only the location of a multivariable zero is important but also its direction. Figure 6 shows the zero location and its directionality when the setting valves are changed.

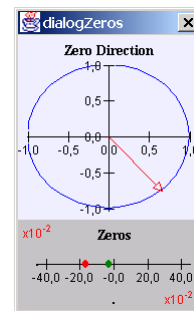


Figure 6: The zeros direction

**The importance of interactivity.** We end this section by stressing that we considered specially necessary to include so many different types of visualizations and to provide a very high level of interactivity. This is convenient, in the first place, because the system increases its pedagogical capabilities, having many parameters that can be easily configurable, both numerically and geometrically. But it also adheres to a greater extent to the best-practices of science education as recommended by specialists (see for instance (Lederman, [17]) and (Jimoyiannis, A. and Komis, V. [16]).

In particular, it lets the students isolate and manipulate parameters thus developing a better understanding of the relationships between physical concepts, variables and phenomena. It also employs a variety of representations that help to understand the underlying relations and processes.

## 5 Conclusions

Control education is going through a period of rapid transition; one of the ultimate objectives is to increase the educational efficiency. In most of the traditional courses on control engineering, has been mainly conducted by using

basic theoretical tutorials. The combination of an interactive environment plus animation brings visualization to a new level and aids in learning and active participation by students. In this paper a simulation model and an animation of a quadruple tank using *Easy Java Simulations* (Ejs) has been presented. Ejs, is a software tool that helps create, in a very easy way, dynamic, interactive scientific simulations in Java language.

The role of this new interactive computer learning experience in control engineering curriculum are two-fold: 1) To provide a new method for delivering classroom material whereby real-world control system engineering concepts are introduced via interactive package, and 2) To provide an opportunity innovative laboratory assignments where students can analyze, design and modify control engineering systems via interactive tools. In conclusion, this educational system can give an efficient way for undergraduate (or graduate) students to understand and test the main features of process dynamics and control using a virtual lab. The Ejs package and the four-tank example is available for free at <http://fem.um.es/Ejs>.

## Acknowledgements

Authors would like to acknowledge CICYT for partially funding this work under grant DPI 2001-1012.

## References

- [1] Andersson, M.; Glad, T.; Norrlöf, M.; Gunnarsson, S.: (2002). A simulation and animation tool for studying multivariable control. *15th World Congress of IFAC, Barcelona 21-26 july*.
- [2] Åström, K. J.; Lundh, M.: (1992) Lund control program combines theory with hands-on experience. *IEEE Control Systems Magazine, vol 12, n° 3, pp. 22-30*.
- [3] Åström, K. J.; Wittenmark, B.: (1997) Computer-controlled systems: Theory and design. 3<sup>rd</sup> ed. *Prentice Hall*.
- [4] Dai, L.; Åström, K.J.: (1999). Dynamic matrix control of a quadruple-tank process. *14th World Congress of IFAC, Beijing*
- [5] Dormido, S.: (2002). Control Learning: Present and Future. Plenary lecture. *Plenary Papers, Survey Papers and Milestones, 15th IFAC World Congress, Barcelona 21-26 july, pp. 81-103*.
- [6] Dormido, S.; Gordillo, F.; Dormido-Canto, S.; Aracil, J.: (2002). "An interactive tool for introductory nonlinear control systems education", *15th. IFAC World Congress, Barcelona 21-26 july*.
- [7] Dormido, S.: (2003). The role of interactivity in control learning. Plenary lecture. *15th IFAC Symposium on Advances in Control Education, Oulu 16-18 june, Preprints, pp. 11-22*.
- [8] Dormido, S.; Berenguel, M.; Dormido-Canto, S.; Rodríguez, F.: (2003). Interactive learning of constrained generalized predictive control. *15th IFAC Symposium on Advances in Control Education, Oulu 16-18 june, Preprints, pp. 201-206*.
- [9] Doyle III, F. J.; Meadows, E. S.: (2000). An interdisciplinary control engineering laboratory course. *IFAC/IEEE Symposium on Advances in Control Education, Gold Coast, Australia*.
- [10] Doyle III, F. J.; Troutman, K.; Meadows, E. S.; Gatzke, E. P.; Vadigepalli, R.; Mahadevan, K.; Whitmyre, G.; Stooles, G.; Richards, P.: (2000). Innovative control education using a 4-tank experiment and the WWW. *IFAC/IEEE Symposium on Advances in Control Education, Gold Coast, Australia*.
- [11] Esquembre, E.: (2002). Easy Java Simulations 3.1, <http://fem.um.es/Ejs>
- [12] García Gabín, W.; Camacho, E. F.: (2002). Application of multivariable GPC to a four tanks process with unstable transmission zeros. *IEEE International Symposium on Computer Aided Control System Design, Proc. pp. 645-650, Glasgow 18-20 september*.
- [13] Gatzke, E. P.; Meadows, E. S.; Wang, C.; Doyle III, F. J.: (2000). Model based control of a four-tank system. *Computers and Chemical Engineering, vol 22, pp. 1503-1509*.
- [14] Johansson, K. H.: (1997). Relay feedback and multivariable control. *PhD thesis TFRT-1048, Dept. of Automatic Control, Lund Institute of Technology, Lund, Sweden*.
- [15] Johansson, K. H.: (2000). The quadruple-tank process: A multivariable laboratory process with an adjustable zero. *IEEE Trans. on Control Systems Technology, vol 8, n° 3, may, pp. 456-465*.
- [16] Jimoyiannis, A.; Komis, V.: (2001). Computer simulations in physics teaching and learning. *Computers & Education, vol 36, pp. 183-204*.
- [17] Lederman, L. M. (1998). ARISE: American Renaissance in Science Education. *FERMILAB-TM-2051*.
- [18] Salt, J.; Albertos, P.; Dormido, S.; Cuenca, A.: (2003). An interactive simulation tool for the study of multirate sampled data system. *15th IFAC Symposium on Advances in Control Education, Oulu 16-18 june, Preprints, pp. 45-50*.
- [19] Sánchez, J.; Morilla, F.; Dormido, S.; Aranda, J.; Ruipérez, P.: (2000). "Conceptual learning of control by Java-based simulations", *IFAC/IEEE Symposium on Advances in Control Education, Gold Coast, Australia*.
- [20] Sánchez, J.; Morilla, F.; Dormido, S.; Aranda, J.; Ruipérez, P.: (2002). "Virtual control lab using Java and Matlab: A qualitative approach", *IEEE Control Systems Magazine, 22, n° 2*.
- [21] Tan, N.; Atherton, D. P.; Dormido, S.: (2003). Systems with variable parameters; classical control extensions for undergraduates. *15th IFAC Symposium on Advances in Control Education, Oulu 16-18 june, Preprints, pp. 287-292*.
- [22] Vadigepalli, R.; Gatzke, E. P.; Doyle III, F. J.: (2001). Robust control of a multivariable experimental four-tank system. *Industrial Engineering Chemical Research, vol 40, pp. 1916-1927*.