Magnetic Levitation System:
A Virtual Lab in "Easy Java Simulation"

S. Dormido, C. Martín, R. Pastor, J. Sánchez, F. Esquembre

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 magnetic levitation system-, 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.

I. INTRODUCTION

Control education is a multi-faceted field. First of all, a set of fundamental concepts that could be called "control principles" (dynamic system, stability, feedback and dynamic compensation) should be introduced to the students. It is also important that students get a deep understanding of the behaviour of feedback systems, a topic that can be quite complex. Finally, from a practical point of view, it is also desirable that students obtain a sound knowledge of analysis and design techniques and about system implementation as well [4].

This wide range of objectives makes control education a non trivial activity. Many of these tasks, in particular those related to mathematical aspects, can be taught properly by means of traditional lectures [3]. However, because of the applied nature of this discipline, labs are indispensable for students to understand many of the subtle details hidden in the study of dynamic systems [2, 9, 15, 24].

Due to great progress on computer hardware and software, it is now possible to provide students with new tools oriented to the control education process. These are interactive programs that can be considered half-way

between regular labs and lectures. The main idea is to have a *multiple-view* representation of a given dynamic system, and some of its attributes, on the computer screen. These views can then be manipulated directly while keeping the coherence of the representation. These tools, called *interactive virtual labs*, can be used to explain basic concepts, to provide new perspectives of a problem, and to illustrate analysis and design topics [1, 5, 7, 19, 23].

Currently, one of the actual problems in control education is the shortage of money and space to create and maintain laboratories. Due to this, many developments in virtual and remote labs through the Internet have received great attention in the last few years. This progress is particularly important in control teaching at distance universities where on-site labs are more difficult [21].

Making use of computer simulations in an instructional context means using the computer to build theoretical models or to model real-world phenomena in order to help students gain insight into the behavior of complex systems. Interacting with an instructional simulation can help learners 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 associate actions with effects, which will lead to better understanding.

Furthermore, the student has an excellent opportunity to "experiment" with his or her own ideas in terms of engineering design through simple interaction with the tool. *Dynamic pictures* can not only be effective in presenting difficult engineering concepts in the classroom, but also beneficial in extending students' experience in analysis and design assignments [16]. This invitation for creativity can be most useful when it comes to specialized control engineering student projects.

In this context, the magnetic levitation (*maglev*) system [12, 18, 22] has received great attention because it presents intricate and interesting properties in both control education and research. The *maglev* system not only exhibits, in an elegant and simple way, complex dynamics, but it also has many relevant applications, such as high-speed transportation systems and precision bearings. From an educational viewpoint, this process is highly motivating and

suitable for laboratory experiments and classroom demonstrations, as reported in the control education literature. The *maglev* system can be quickly set up in the open loop stable and unstable (repulsive and attractive fields) configuration. By adding a second magnet, the system is transformed in a multivariable control problem with the level of interaction which depends on the distance between both magnets.

This paper is organized as follows. In Section 2 we briefly describe the magnetic levitation system. *Easy Java Simulation*, a software package that supports the creation of interactive dynamic simulations in Java, is introduced in Section 3. In Section 4 we discuss the implementation of the magnetic levitation system virtual lab. Section 5 describes a particular configuration working mode of the levitation system control. Finally, in Section 6 we present some conclusions.

II. MODEL OF THE MAGNETIC LEVITATION SYSTEM

The system modelled in this work is the ECP magnetic levitation system (model 730) [10]. Figure 1 shows the free-body diagram of two suspended magnets. Each magnet is acted on by forces from either driver coil, from the other magnet, from gravity and from friction (modelled as viscous). The differential equations for both magnets are given by:

$$m_1 \ddot{y}_1 + c_1 \dot{y}_1 + F_{m12} = F_{u11} - F_{u21} - mg$$
$$m_2 \ddot{y}_2 + c_2 \dot{y}_2 - F_{m12} = F_{u22} - F_{u12} - mg$$
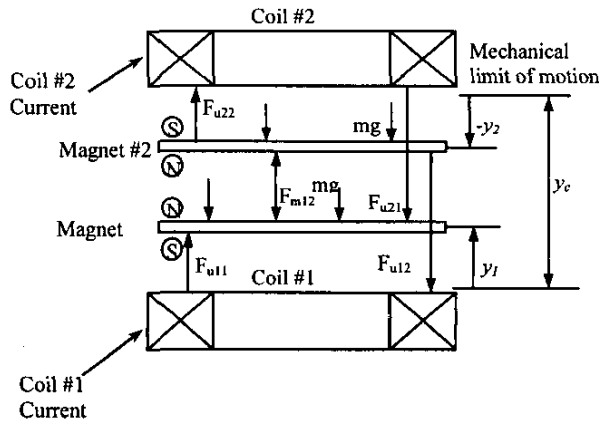

Fig. 1. Free body diagram

The magnetic force terms are modeled by [10]:

$$F_{u11} = \frac{i_1}{a(y_1 + b)^N} \quad (1)$$

$$F_{u12} = \frac{i_1}{a(y_c + y_2 + b)^N} \quad (2)$$

$$F_{u21} = \frac{i_2}{a(y_c - y_1 + b)^N} \quad (3)$$

$$F_{u22} = \frac{i_2}{a(-y_2 + b)^N} \quad (4)$$

$$F_{m12} = \frac{c}{(y_{12} + d)^N} \quad (5)$$

$$y_{12} = y_c + y_2 - y_1$$

where a, b, c, d and N are constants which may be determined by numerical modelling of the magnetic configuration. Typically $3 < N < 4.5$. In this case, we have taken $N = 4$ and a, b, c and d have been determined by empirical methods in order to adjust the simulation and experimental results.

The identification of the parameters of the model was done by measuring the specific input/output characteristics of the magnet/coil actuators and the magnet/magnet interactions as they vary with relative position. The strong magnetic field non-linearity, however, are inherent in this class of magnetic systems. For most control design purposes, the upper and lower actuators' characteristics may be assumed to be identical.

The interaction of the two magnets has a nonlinear force characteristic. When two magnets are used in the *maglev*, they are usually placed with like poles facing each other so that they repel. When the lower magnet is stationary, the upper magnet comes to rest in equilibrium between the upward repulsive magnet force and downward gravitational force. The magnetic force characteristic may be measured by adding and subtracting weight from the upper magnet and measuring the equilibrium height.

Using this procedure, we obtained the following values for the parameters: $a = 1.65$ N, $b = 6.2$ cm, $c = 2.69$ Ncm^3 y $d = 4.2$ cm.

III. 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:

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) Model: 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) Control: defines certain actions that a user can perform on the simulation.

3) View: shows a graphical representation of the different states that the phenomenon can have. This representation can be done in a realistic or schematic form.

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.

Thus, finally, creating a simulation with *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. Teachers can create the same simulation with different views for different objectives or for 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 author. 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[1] that properly describe the system

2. Initialize, in a correct way, these variables

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.

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 [20,21].

IV. IMPLEMENTATION OF THE MAGLEV WITH EJS

This section briefly describes the main features of the *maglev* simulation we created using *Ejs*. When developing this type of simulations, one of the most important things that the teacher needs to have in mind is the correct configuration of windows and operating controls of the simulation that will facilitate the student's understanding of the virtual lab (Dormido, [6]). In our case, this view (using *Ejs* terminology) is divided into several elements as displayed in Figure 2.

[1] The word *variables* means here computer variables, which may correspond either to constants, parameters or state variables in physical terminology.

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

A second requirement was to provide a set of interactive elements that could be manipulated by the students to make changes in a dynamic way during an experiment.

The view is therefore structured in one main window and three secondary dialog windows. The main window is located in the left-hand side of Figure 2. In its upper part we schematically display the *maglev*. We have tried to make this first image as self-explanatory as possible. Set points can be changed interactively by dragging the arrows up and down.

The lower part of the main window is filled with buttons and check boxes that allow the user to choose the operating conditions of the system. With these elements the user can select to:

- *Play*, *pause* and *reset* the simulation.

- Choose the control system mode: *manual* or *automatic*.

- Change the mass of the magnets (*change m*).

- Change the magnets polarity (*polarity m#1* and *polarity m#2*).

- Select with those magnets that it will operate the *maglev* (*upper magnet* and/or *lower magnet*).

The last two options can be used to configure the *maglev* virtual lab as a variety of single-input single-output (SISO) and multi-input multi-output (MIMO) systems. By using a repulsive force from the lower coil to levitate a single magnet, an open loop stable SISO system is created. Attractive levitation via the upper coil effects an open loop unstable system. Two magnets may be raised by a single coil to produce a SIMO plant. If two coils are used, a MIMO plant is produced. These may be locally stable or unstable depending on the selection of the magnet polarities and the nominal magnet positions. The plant has inherently strong nonlinearities due to the natural properties of magnetic fields. Thus, this virtual lab provides a dynamically rich test bed for implementation of introductory through advanced control methods. Figure 3 shows four possible different operating conditions.

The interaction with the virtual lab offers more possibilities than with the real system, since actions such as moving a magnet can be carried out graphically, by dragging it with the mouse to the desired new position.
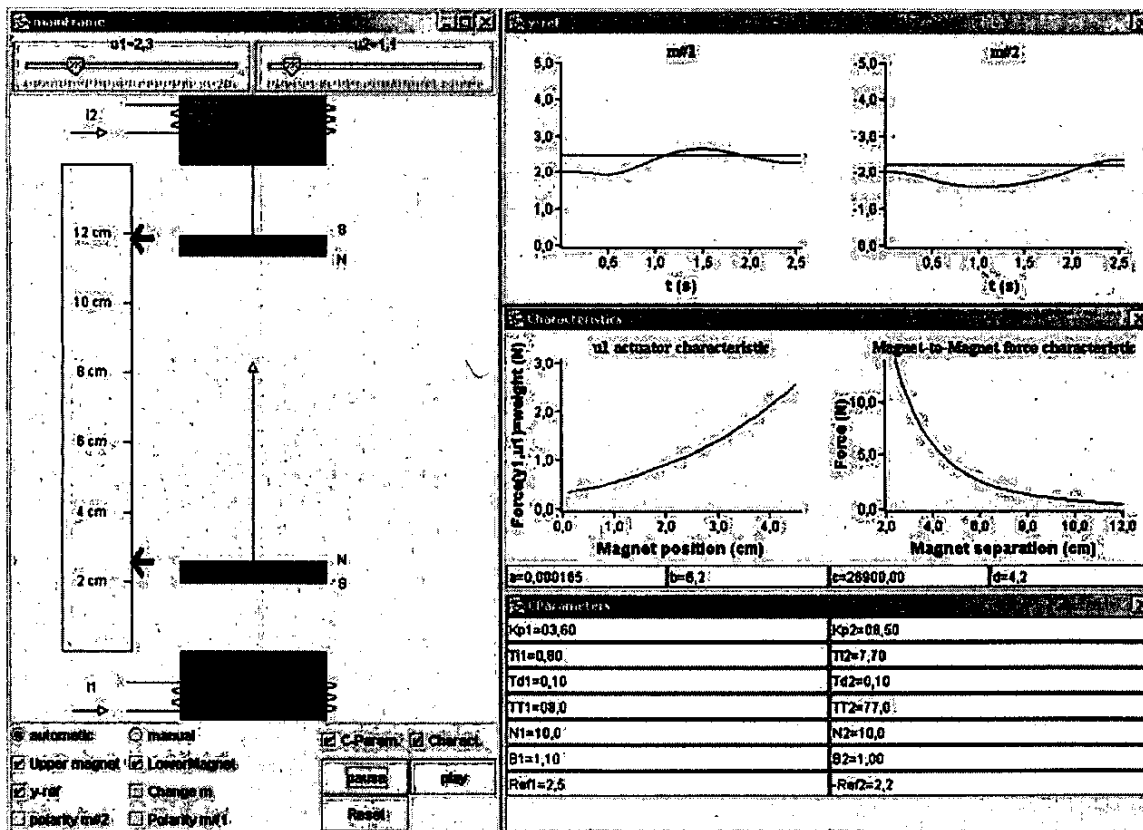

Fig. 2. Magnetic levitation virtual lab

This virtual lab also allows the interactive change of parameters and characteristics of the system, such as its mass, for instance, so that we can simulate disturbances. Thus, if we check the box[2] labelled *change m*, a small circle appears on the left lower corner of both magnets. Dragging these circles will modify the width of the magnet, which in turn proportionally affects its mass. A second interesting modification consists in changing the polarity of both magnets (*polarity m#1* and *polarity m#2*). You will appreciate in Figure 2 that there is a blue arrow associated to each magnet whose module is proportional to the force acting on it. This helps us locate the equilibrium points of the system.

The three dialog windows on the right-hand part of Figure 2 (*y-ref, Characteristics* and *CParameters*) are shown by checking the corresponding box in the main window (*y-ref, Charact.* and *C Param* , respectively). The first one- *y-ref* - shows, in the right upper window, both the set point (red line) and the controlled variable (blue line). The second -*C Param*- allows changing the parameters of the characteristic curves of the actuator (*a* and *b*) and the interaction between both magnets (*c* and *d*) and observe how the corresponding curves change. Finally, the third window shows the parameters of the two PID controllers, with two degrees of freedom, that provide the control for the *maglev* in a decentralized control structure. The parameters of both controllers can be changed interactively during a simulation.

A. The importance of interactivity.

We end this section by stressing that we considered particularly important 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 having many parameters that can be easily configurable, both numerically and geometrically, increases the system's pedagogical capabilities.

a) Repulsive levitation. Open loop stable

b) Attractive levitation. Open loop unstable

Checkmark controls are graphic elements whose state can be set on or off individually.

c) MIMO levitation, locally open loop stable
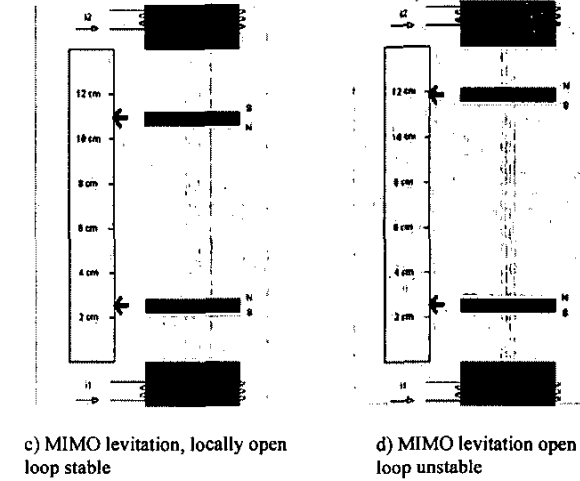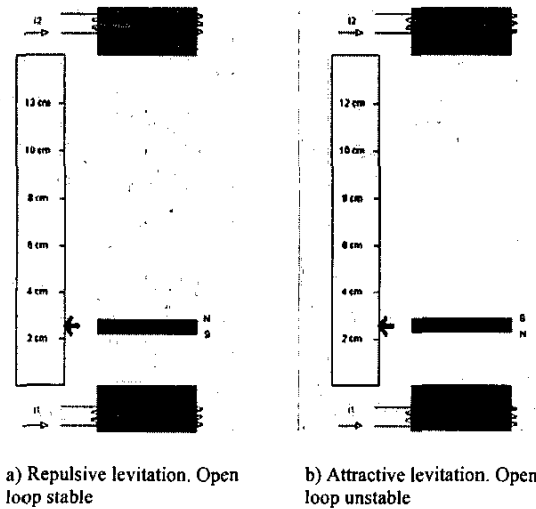
d) MIMO levitation open loop unstable

Fig. 3. *Maglev* virtual lab configurations

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. [14]). In particular, it lets the students isolate and manipulate parameters, thus developing a better understanding of the relationships between physical concepts, variables and phenomena.

V. CONTROL OF THE MAGLEV VIRTUAL LAB.

We now describe one of the control experiences that we carried out with our *maglev* virtual lab.

A. Decentralised PID Control

The equations for the linearized system around its equilibrium point (u_{i0}, y_{i0}), $i = 1, 2$ are:

$$\ddot{y}_i^* + \frac{k_i'}{m} y_i^* = \frac{k_{u_i}'}{m} u_i^* , \ u_i^* = u_i - u_{i0} , \ y_i^* = y_i - y_{i0} \quad (6)$$

$$k_{ui}' = \frac{1}{a(y_{i0} + b)^4} , \ k_i' = \frac{4u_{i0}}{a(y_{i0} + b)^5}$$

$$u_{i0} = a(y_{i0} + b)^4 \left[\frac{4c}{(y_{120} + d)^4} + mg \right]$$

Take $y_{10} = 2$ cm, $y_{20} = -2$ cm, and $m = 120.3$ g. Taking Laplace transform of (6) for $i = 1,2$ we obtain:

$$\frac{\dot{y}_1^*}{u_1^*} = \frac{11.133}{s^2 + 15.4931} \quad \frac{\dot{y}_1^*}{u_1^*} = \frac{11.133}{s^2 - 1.0672}$$

Figure 4 show the results for the following control parameters: $K_{p1} = 3.6$, $T_{i1} = 0.5$, $T_{d1} = 0.1$, $T_{i1} = 8$, $N_1 = 10$, $b_1 = 1.1$ and $K_{p2} = 8.43$, $T_{i2} = 1.5$, $T_{d2} = 0.2$, $T_{i2} = 77$, $N_2 = 10$, $b_2 = 1$.

VI. CONCLUSIONS

Control education is going through a period of rapid transition; one of the ultimate objectives is to increase the educational efficiency. Most of the traditional courses on control engineering are based on theoretical tutorials.

a) Changing set point for magnet 1

b) Changing set point for magnet 2

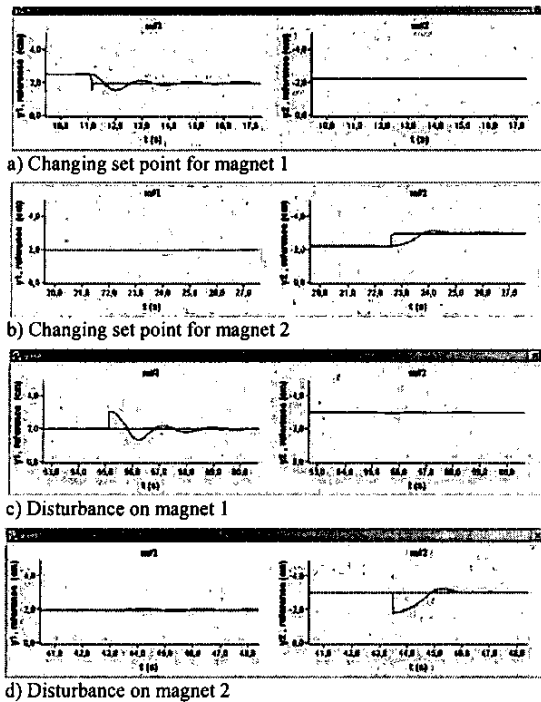c) Disturbance on magnet 1

d) Disturbance on magnet 2

Fig. 4. *Maglev* in a MIMO configuration

The combination of an interactive environment plus animation opens new visualization possibilities that improve the learning and facilitates a more active role of the students [1], [13]. In this paper, we have introduced a simulation model and an interactive animation of a magnetic levitation system that we have created using *Easy Java Simulations*. *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 packages, and 2) To provide an opportunity for innovative laboratory assignments where students can analyze, design, and modify control engineering systems via interactive tools. In conclusion, this educational system can provide a more efficient way for undergraduate (but also graduate) students to understand and test the main features of dynamic processes and control using a virtual lab. The *Ejs* package and the magnetic levitation system example are available for free at *http://fem.um.es/Ejs*.

REFERENCES

[1]   M. Andersson, T. Glad, M. Norrlöf, S. Gunnarsson, "A simulation and animation tool for studying multivariable control", *15th World Congress of IFAC*, Barcelona, july, 2002.

[2]   K. J. Åström, M. Lundh, "Lund control program combi-nes theory with hands-on experience", *IEEE Control Systems Magazine*, vol 12, n° 3, 1992, pp 22-30.

[3]   K. J. Åström, B. Wittenmark, *Computer-controlled systems: Theory and design*. 3rd ed. Prentice Hall; 1997.

[4]   S. Dormido, "Control Learning: Present and Future", Plenary lecture. *Plenary Papers, Survey Papers and Milestones, 15th IFAC World Congress*, Barcelona, july, 2002, pp. 81-103.

[5]   S. Dormido, F. Gordillo, S. Dormido-Canto, J. Aracil, "An interactive tool for introductory nonlinear control systems education", *15th. IFAC World Congress*, Barcelona, july, 2002.

[6]   S. Dormido, "The role of interactivity in control learning", Plenary lecture. *15th IFAC Symposium on Advances in Control Education*, Oulu, June, 2003, Preprints, pp. 11-22.

[7]   S. Dormido, M. Berenguel, S. Dormido-Canto, F. Rodríguez, "Interactive learning of constrained generalized predictive control", *15th IFAC Symposium on Advances in Control Education*, Oulu, June, 2003, Preprints, pp. 201-206.

[8]   S. Dormido, F. Esquembre, "The quadruple-tank process: An interactive tool for control education", *European Control Conference*, Cambridge, September, 2003, Preprints, pp. 201-206.

[9]   F. J. Doyle III, E. S. Meadows, "An interdisciplinary control engineering laboratory course", *IFAC/IEEE Symposium on Advances in Control Education*, Gold Coast, Australia, 2000.

[10]   ECP Educational Control Products: Manual for Model 730 (1999). *Magnetic Levitation System*

[11]   F. Esquembre, Easy Java Simulations 3.1, *http://fem.um.es/Ejs*, 2002.

[12]   R. K. Galvao, T. Yoneyama, F. M. de Araujo, R. G. Machado, "A simple technique for identifying a linearized model for a didactic magnetic levitation system", *IEEE Transactions on Education*, vol. 46, n° 1, 2003, pp. 22-25.

[13]   E. P. Gatzke, E. S. Meadows, C. Wang, F. J. Doyle III, "Model based control of a four-tank system". *Computers and Chemical Engineering*, vol. 22, 2000, pp. 1503-1509.

[14]   A. Jimoyiannis, V. Komis, "Computer simulations in physics teaching and learning", *Computers & Education*, vol. 36, 2001, pp. 183-204.

[15]   K. H. Johansson, "The quadruple-tank process: A multivariable laboratory process with an adjustable zero", *IEEE Trans. on Control Systems Technology*, vol. 8, n° 3, may, 2000, pp. 456-465.

[16]   M. Johansson, M. Gäfvert, K. J. Åström, "Interactive tools for education in automatic control", *IEEE Control Systems Magazine*, vol. 18, n° 3, 1998, pp. 33-40.

[17]   L. M. Lederman, "ARISE: American Renaissance in Science Education", *FERMILAB-TM-2051*, 1998.

[18]   S. H. Lee, H. K. Sung, J. T. Lim, Z. Bien, "Self-tuning control of electromagnetic levitation systems", *Control Engineering Practice*, vol. 8, 2000, pp. 749-756.

[19]   J. Salt, P. Albertos, S. Dormido, A. Cuenca, "An interactive simulation tool for the study of multirate sampled data system" *15th IFAC Symposium on Advances in Control Education*, Oulu, June, 2003, Preprints, pp. 45-50.

[20]   J. Sánchez, F. Morilla, S. Dormido, J. Aranda, P. Ruipérez,. "Conceptual learning of control by Java-based simulations", *IFAC/IEEE Symposium on Advan-ces in Control Education*, Gold Coast, Australia, 2000.

[21]   J. Sánchez, F. Morilla, S. Dormido, J. Aranda, P. Ruipérez, "Virtual control lab using Java and Matlab: A qualitative approach", *IEEE Control Systems Magazine*, vol. 22, n° 2, 2002, pp. 8-20.

[22]   P. S. Shiakokas, D. Piyabongkarn, "Development of a real-time digital control system with a hardware-in the-loop magnetic levitation device for reinforcement of control education", *IEEE Transactions on Education*, vol. 46, n° 1, 2003, pp. 79-87.

[23]   N. Tan, D. P. Atherton, S. Dormido, "Systems with variable parameters; classical control extensions for undergraduates", *15th IFAC Symposium on Advances in Control Education*, Oulu, June, 2003, Preprints, pp. 287-292.

[24]   R. Vadigepalli, E. P. Gatzke, F. J. Doyle III, "Robust control of a multivariable experimental four-tank system", *Industrial Engineering Chemical Research*, vol. 40, 2001, pp. 1916-1.