Proceedings

2007 IEEE International Symposium on Industrial Electronics

June 4-7, 2007

Centro Cultural and Centro Social Caixanova - Vigo, Spain

Main
Committees
Messages
- General Chairs
- Program Chairs

Table of Contents
Author Index
Reviewers

Technical support:
isie07@dte.uvigo.es

2007 IEEE International Symposium on Industrial Electronics

Centro Cultural and Centro Social Caixanova
Vigo - SPAIN
June 4-7, 2007

PROCEEDINGS

IEEE Catalog Number: 07TH8928C
ISBN: 1-4244-0755-9
Library of Congress: 2006935487

Remote Control Laboratory Using Matlab and Simulink

L.M. Jiménez, R. Puerto, O. Reinoso, R.P. Ñeco, C. Fernández
Universidad Miguel Hernández (Spain)
Email: luis.jimenez@umh.es

***Abstract*—In this paper, a teaching application for remote real-time execution of physical process controllers is presented. This application has been developed using the platform Matlab/Simulink. The motivation of this work is based on the little availability of real physical systems or laboratories to perform the experiments in control courses. In this way, control lab assignments with various physical processes present in the remote laboratories can be performed. Also, some examples that show the validity and applicability of the presented architecture are introduced.**

I. INTRODUCTION

The work presented in this paper pretends to provide a general architecture for remote execution in real time of physical processes using the Matlab platform. The main motivation is, on one hand, the lack of lab scale models or real physical systems, and on other, the little availability of schedules in the laboratories where the students of automatics and process control subjects develop the practical assignments corresponding with the theoretical contents. Therefore, the presented application is focused on the real-time remote control of a physical system through Internet and Matlab as teaching and development platform. This application will allow the students to simulate the operation of a controller for a certain physical process using Internet, and the control in real time of the physical process using the designed controller. In this way, the student is allowed to develop these activities without being in the laboratory where the real system is installed. Also, the application returns to the user all the information relating the performed execution, besides graphic and other data of interest.

During last years, Internet has provided a set of possibilities that allow carrying out diverse tools increasing the quality in the learning and teaching processes. Distance learning through the use of remote laboratories via Internet is a highly topical issue as a consequence of the large potential to increase the quality in the process of teaching and learning [1].

A distance laboratory allows users to perform experiments from a remote location. Distance laboratories can be divided into two classes: virtual labs and remote labs [2]. In the first ones the students can run simulations remotely with possible animations. Some laboratory skills, such as the statistical analysis of data, can be learned outside of the laboratory. However, experimental design can only be learned from using real equipment in real experiments, often through a certain amount of trial and error. On the other hand, remote labs are laboratories where students can interact with actual experiments via Internet. A more detailed study between remote and virtual labs is presented in [3]. Without a doubt, the use of real physical systems allows the students to acquire knowledge in a more efficient way than only using simulated exercises.

Different real and virtual experiments through remote laboratories have been proposed in a large number of disciplines [4][5]. Particularly, in the automatic control discipline, practical experiments with physical systems are significantly important [6]. As a consequence, several distance control laboratories have been proposed recently allowing to students the use of the systems in such laboratories [7][8]. Most of the experiences presented combine different developments that allow make real experiments on control systems through Internet [8][9].

One of the main features of the system proposed in this paper consists of using a set of tools extensively known by the students of automatic control such as Matlab and Simulink. The development of computer aided control system design (CACSD) tools based on Matlab environment, the Simulink toolbox and the Real-Time Workshop toolbox is extensively analyzed in [10]. These tools enable educators and students to focus on control systems design, implementation, and evaluation rather than on time-consuming, low-level programming.

Schmid [9] presents a virtual laboratory which uses Matlab/Simulink for simulations using virtual reality. Interactive animation of control engineering experiments is performed using VRML models of laboratory plants. Bonivento et al. proposed a remote control laboratory based on Matlab/Simulink but with an specific software application developed in Visual C++ [11] to interact with the physical models. In [12] the Matlab Web Server tool allowing the remote execution of Matlab non real-time code using a web browser is detailed. Finally Overstreet et al. [13] shows a platform where the user can run an experiment using a controller which resides not only on the server but also on the client, compiling and executing it on the user machine.

1-4244-0755-9/07/$20.00 ©2007 IEEE

We have chosen the Matlab/Simulink platform (with additional toolboxes) as the development tool of the application proposed in this paper for the following reasons: first of all, Matlab, Simulink and additional toolboxes constitute a reliable platform widely used with an adequate technical support and an extensively use in control applications. Second, it is possible to develop an application in the laboratory in a lesser time than with other tools or platforms. Also, Matlab provides several tools for the remote execution of programs and a real time control toolbox to manage a physical system through a data acquisition system.

The remainder of the paper is organized as follows. In the following section a general description of the architecture is introduced. Section III presents two physical systems connected to the remote laboratory (called RECOLAB) with some examples of control design. Section IV shows the experience of student usage. Finally, section V presents the conclusions.

II. General Description Of The System

The main goal on developing the system was to achieve accurate real-time executions over physical systems through internet, transparently connecting parameterized Simulik schemes with user's control design. To do this, it has been necessary to integrate a specific hardware and software architecture.

The general scheme of the application architecture is shown in Fig. 1. In this diagram, the hardware and software elements are split in two main blocks: local area where the user works, and remote area where the whole physical system and control elements are located. The detailed elements of local and remote area are the following:

Local area:

- Computer with Internet connection and an HTTP 4.0 client application. The application is optimized for Internet Explorer 6, Netscape 7 and Firefox 1.x with a minimum resolution of 800x600.

Remote area:

- High speed Internet Connection.
- Computer Server: the current implemented system consists of a PC Pentium III running Microsoft Windows 2000 operating system.
- Data acquisition system: NI 6024E acquisition board with analogue and digital I/O.
- Physical system to control: Two models are currently implemented, a DC motor model 33-002 from Feedback and an airflow slider cylinder.
- Image capture system and web video server: an Axis network camera with MPEG-4 video compression streams video and still images to the user.
- Http Server Apache v.2.0.54 with PHP 5.0 module. This server allows the communication of the computers using the http protocol.
- MATLAB R12 with SIMULINK V. 4.1: executes the program that makes possible the real-time control of the system and the generation of the results in a file.
- Real -- Time Windows Target Toolbox V. 2.1: this toolbox allows executing Simulink schemes in real time. For this purpose, it provides the necessary blocks for the interaction with the data acquisition system.

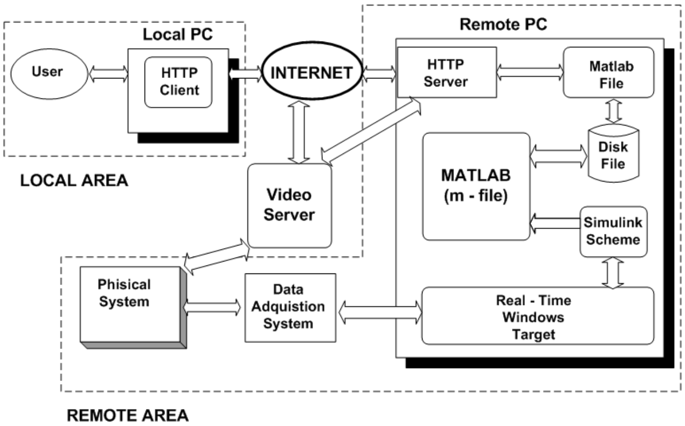


Fig. 1. General architecture of RECOLAB system.

The Software part of the system has two modules clearly differentiated:

1. *Web application*: this includes client-server communication using HTTP/HTML protocol, the user interface, user's access control, and the main CGI application. The Common Gateway Interface (CGI) is a standard for interfacing external applications with information servers, such as HTTP or Web servers. This application is in charge of resource access and communication between Apache and Matlab.

2. *Real-time application*: this is a set of predefined Simulink control schemes and Matlab code, based on Real time Windows Target toolbox, which implements the real-time execution of Simulink schemes over a specific physical system.

Different programming languages and development tools has been used for each part. The non-critical task such as user interface, security access and resource sharing, have been coded in PHP (v.5.0) running over an Apache HTTP server. PHP is a popular script language that has been chosen as far as it is an open language widely supported by most web servers and O.S. platforms, and with an extensive library that supports every network protocol and data base access. PHP code runs on the web sever so it shows a controlled environment for the programmer and can communicate with any other process running in the server (Matlab application in our case).

Of course, PHP as a scripting language is not suitable for real-time applications. The hard real-time core of RECOLAB (feedback control of physical systems) is developed in Simulink and compiled with the Real-Time Windows Target tool.

Fig. 2 shows the functional flow graph of main system application. A more comprehensive description of implementation can be found in [8].

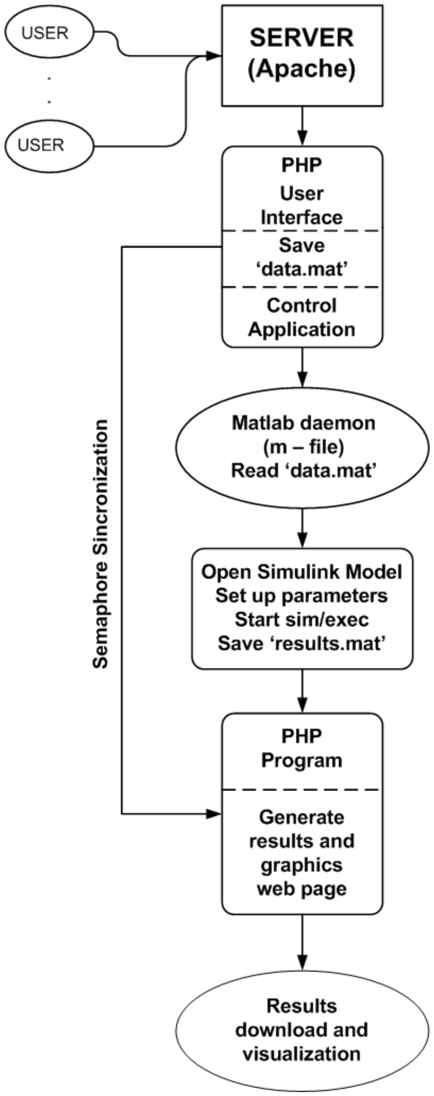

USER
USER
SERVER (Apache)
PHP User Interface
Save 'data.mat'
Control Application
Semaphore Sincronization
Matlab daemon (m – file) Read 'data.mat'
Open Simulink Model Set up parameters Start sim/exec Save 'results.mat'
PHP Program
Generate results and graphics web page
Results download and visualization

Fig. 2. Functional description of software application

III. Experimental Systems And Examples

This section describes the physical systems connected to RECOLAB and some examples. We only describe real-time examples (not simulation) since this is the most interesting feature of RECOLAB. Currently, the laboratory has two physical systems connected to RECOLAB, which allows performing a wide range of control experiments. These systems are a DC engine and a sliding cylinder Fig. 3.

The DC engine consists of a Feedback 33-100 mechanical unit. The mechanical components of this unit are: the DC engine itself, an analog tachometer, a position potentiometer, absolute and incremental digital encoders, and a magnetic brake. Fig. 3.a shows this system.

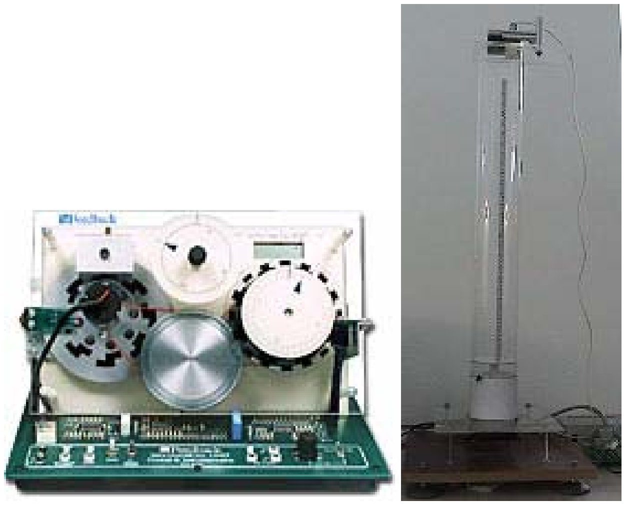


Fig. 3. (a) Feedback 33-100 mechanical unit, (b) Sliding cylinder

The sliding cylinder consists of a tube made of methacrylate where an object slides propelled by an air flow obtained from a DC fan. A photoelectric sensor placed in the top of the tube obtains the position of the object. Fig. 3.b shows an image of the sliding cylinder.

When a user enters to the system, he/she is allowed to perform a simulation or a real-time execution for each one of the systems mentioned above. The different types of experiments and control schemes applicable to each system are the following:

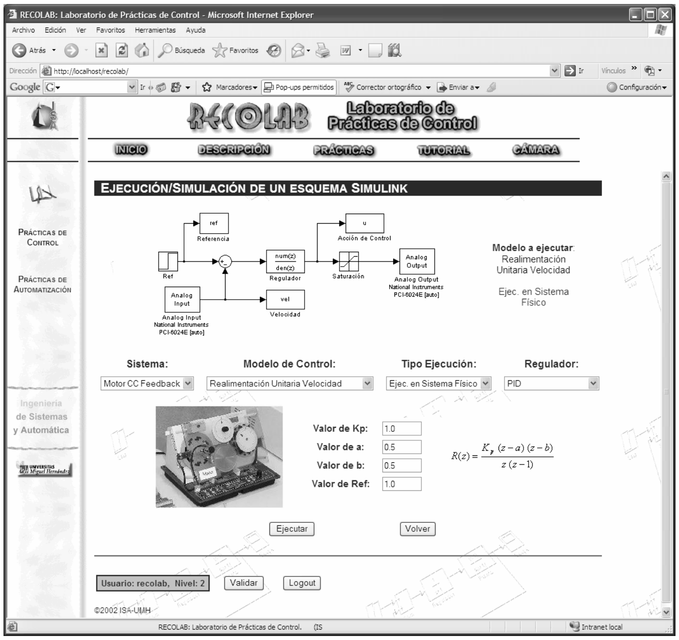
(a) DC engine
- System identification
- PID controllers for velocity and position control
- Algebraic controllers based on poles placement technique
- State feedback control with and without reference following

(b) Sliding cylinder
- PID controllers

- Algebraic controllers based on poles placement technique
- Algebraic controllers based on minimum time and finite time
- Minimum variance controller with and without integral action
- State feedback control
- Predictive control
- Optimal control

As can be seen, the identification experiment is only allowed for the DC engine, whereas this is not the case for the sliding cylinder because of its complexity. However, this complexity allows on the other hand performing more complex controllers which have no sense in the DC engine case.

Once the user has accessed to the system, a page appears in which all needed data to perform the real-time execution is requested. User can choose the physical system (DC motor, sliding cylinder), the control model (speed or position feedback, space state feedback, identification …), the type of execution (Simulation or Real-time execution) and Regulator type (PID, algebraic regulator, state matrix gain, …). After introducing all data, the experiment can be performed, and when it is finished, the output signal is shown in the screen. Moreover, the application allows to download a `.mat' file with the values of the most significant signals (output, control action, etc.) in order to be analyzed by the user. Fig. 4 shows an example of velocity DC control using a PID controller.


RECOLAB: Laboratorio de Prácticas de Control - Microsoft Internet Explorer
Archivo Edición Ver Favoritos Herramientas Ayuda
Atrás
Búsqueda
Favoritos
Dirección
Ir
Vínculos
Google
Marcadores
Pop-ups permitidos
Corrector ortográfico
Enviar a
Configuración
RECOLAB
Laboratorio de Prácticas de Control
INICIO
DESCRIPCIÓN
PRÁCTICAS
TUTORIAL
CÁMARA
Prácticas de Control
Prácticas de Automatización
Ingeniería de Sistemas y Automática
Universitas Miguel Hernández
ref
Referencia
u
Acción de Control
Ref
num(z)
den(z)
Regulador
Saturación
Analog Output
Analog Output National Instruments PCI-6024E [auto]
Analog Input
Analog Input National Instruments PCI-6024E [auto]
vel
Velocidad
Ejecutar
Volver
Validar
Logout
Intranet local

Dirección: http://localhost/recolab/

EJECUCIÓN/SIMULACIÓN DE UN ESQUEMA SIMULINK

Modelo a ejecutar: Realimentación Unitaria Velocidad

Ejec. en Sistema Físico

Sistema: Motor CC Feedback

Modelo de Control: Realimentación Unitaria Velocidad

Tipo Ejecución: Ejec. en Sistema Físico

Regulador: PID

Valor de Kp: 1.0

Valor de a: 0.5

Valor de b: 0.5

Valor de Ref: 1.0

$$R(z) = \frac{K_p\,(z-a)\,(z-b)}{z\,(z-1)}$$

Usuario: recolab, Nivel: 2

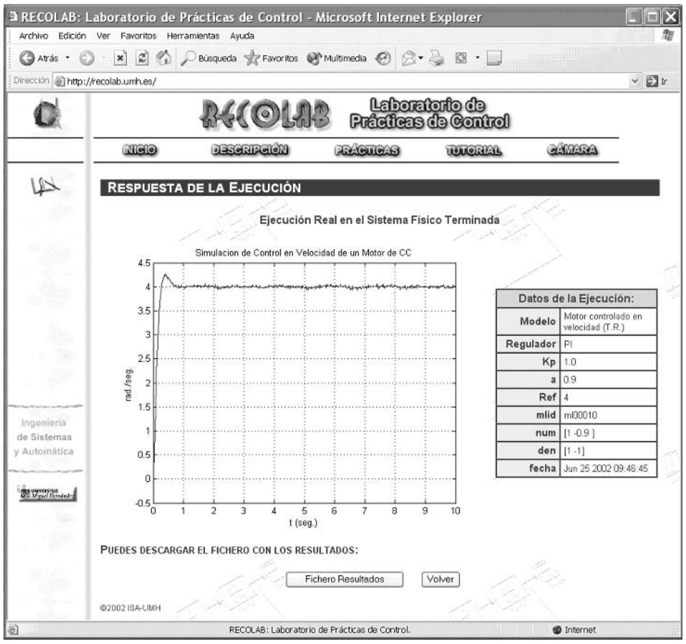
©2002 ISA-UMH

RECOLAB: Laboratorio de Prácticas de Control. (IS

Fig. 4. User interface of RECOLAB. Example of PID controller.

After the execution, a web page with the graph of the output signal is presented to the student (in this case the engine velocity). This page is shown in Fig. 5, which also shows that the system allows the download of all the signals involved in the execution in order to be analyzed by the students.

During real-time execution the user has access to a compressed video stream showing the experiment. For this, a high bandwidth internet access is required.


RECOLAB: Laboratorio de Prácticas de Control - Microsoft Internet Explorer
Archivo Edición Ver Favoritos Herramientas Ayuda
Atrás
Búsqueda
Favoritos
Multimedia
Dirección
Ir
RECOLAB
Laboratorio de Prácticas de Control
INICIO
DESCRIPCIÓN
PRÁCTICAS
TUTORIAL
CÁMARA
Ingeniería de Sistemas y Automática
Universitas Miguel Hernández
Simulacion de Control en Velocidad de un Motor de CC
rad /seg.
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
-0.5
0
1
2
3
4
5
6
7
8
9
10
t (seg.)
Fichero Resultados
Volver
Internet

Dirección: http://recolab.umh.es/

RESPUESTA DE LA EJECUCIÓN

Ejecución Real en el Sistema Físico Terminada

Datos de la Ejecución:	
Modelo	Motor controlado en velocidad (T.R.)
Regulador	PI
Kp	1.0
a	0.9
Ref	4
mlid	ml00010
num	[1 -0.9]
den	[1 -1]
fecha	Jun 25 2002 09:46:45

PUEDES DESCARGAR EL FICHERO CON LOS RESULTADOS:

©2002 ISA-UMH

RECOLAB: Laboratorio de Prácticas de Control.

Fig. 5. Results web page. Example of PID controller.

IV. System Usage Experience

The first RECOLAB protoype started being used in 2003. So far, the response of the students has been satisfactory. First, there have been a high number of accesses to the simulation system; and more importantly, there have also been a high number of accesses to the real time control of the physical equipment.

The system has been used supporting practice sessions in four control courses from the 2003/2004 academic year. There were about 275 students per academic year which could be potential users of the RECOLAB system. Even though these students had attended the practice sessions (it was compulsory, as the system was not fully tested), 73% of them accessed and used the system to repeat remotely some of the practice sessions they have already attended. During the 2004/2005 academic year 87% of the students accessed to the system and used it to repeat some practice sessions. Finally during the last academic year (2005/2006) most of the students were connected to the system. As a conclusion, the students were able to continue working after the laboratory sessions: some of them performed additional experiments with different parameters or different control schemes; while others simply finished the work they have not been able to finish in the laboratory due to time limitations.

The average connection time per session was 35 minutes. Other advantage of the RECOLAB system when compared to a traditional laboratory exercise is the possibility to choose the number and duration of the sessions used to perform a certain experiment with the physical system

The effectiveness of the tool was evaluated by all participants. Table I summarizes the survey results from undergraduate students at the Department of Industrial Systems Engineering at Miguel Hernández University, Spain, in 2005/06. Overall, the survey clearly revealed that the tool received a very positive feedback. However the tool can be improved in partial aspects. Students requested more physical systems were included in the internet laboratory so that they could manage and test control techniques over more physical systems. Also, in some specific occasions some students had connection problems as a consequence of other students were using the remote physical system to test their control schemes, as far only one real-time execution can be done concurrently in the system. In these cases, the access time was higher than they expected.

TABLE I

STUDENT SURVEY RESULTS

Questions	Fully Disagree	Partial Disagree	Partial Agree	Agree	Fully Agree
a. Were the learning procedures clearly defined?	0%	0%	10%	23%	67%
b. Was clear the use of the system?	0%	0%	5%	18%	77%
c. What about the help system?	0%	0%	0%	12%	88%
d. Do allow the system help to understand the importance of practical experiments?	0%	0%	0%	10%	90%
e. Did the use of this tool improve your knowledge in control engineering?	0%	0%	2%	15%	83%
f. Overall, how would you rate this tool?	0%	0%	1%	14%	85%

V. CONCLUSIONS

In this paper, an architecture that allows real time control through the internet has been presented. This architecture is based on Matlab and Simulink that are common tools in engineering degrees of most universities. Besides, the proposed system allows data interchange (results, plots, video, etc.) between the user and the remote laboratory.

The student can choose among predefined values to modify the type of the control scheme (physical system, control layout, regulator, sample time, etc). Such parameterizations are precompiled and allow a reliable real time execution of the process. Besides, it is also possible to add completely new control schemes just by compiling them using the Real-Time Windows Target toolbox.

The main advantage of the proposed system is that it helps the student to perform practice experiments remotely without an strict timetable. Apart from this teaching application, the tool developed and the ideas presented in this paper can also be used to test new control schemes over different physical equipments.

Even though the current version of RECOLAB is operative and working, the system is continuously under development. Some improvements for future versions include: supervision of variables using a new Java client; increase in the number of practice sessions and control schemes available; transmission of compressed video in order to deal with limited bandwidth networks; and addition of control theory tutorials.

REFERENCES

[1] Ausserhofer, "Web-based teaching and learning: A panacea?", IEEE Communications Magazine, 37(3), 92-96 (1999)

[2] M. Casini, D. Prattichizzo and A. Vicino, "The automatic control telelab: A user-friendly interface for distance learning", IEEE Transactions on Education, 46(2), 252-257 (2003)

[3] M. Exel, S. Gentil, F. Michau and D. Rey, "Simulation workshop and remote laboratory: two web-based training approaches for control", Proc. American Control Conference, 3468-3472 (2000)

[4] M. Shor, "Remote-access engineering educational laboratories: Who, what, when, why and how?" Proc. American Control Conference, 2949-2950 (2000)

[5] R. Robson and M. Shor, "A student-centered feedback control model of the educational process", Frontiers in Educational Conference, 2, 14-19 (2000)

[6] N. A. Kleir, K.J. Astrom, D. Auslander, K. Cheok, G.F. Franklin, N. Masten and M. Rabins, "Control systems engineering education", Automatica, 32(2), 147-166 (1996)

[7] R. Pastor, J. Sanchez and S. Dormido, "Related: A framework for publish web laboratory control system", In IFAC Internet Based Control Education, 207-212 (2001)

[8] R. Puerto, L.M. Jimenez, O. Reinoso, C. Fernandez and R. Neco, "Remote control laboratory using Matlab and Simulink: Application to a dc control model", In IFAC Internet Based Control Education (2004)

[9] C. Schmid, "Virtual control laboratories and remote experimentation in control engineering", Proc 11th Annual Conference on Innovations in Education for Electrical and Information Engineering, 213-218 (2001)

[10] W. Dixon, D. Dawson, B. Costic and M. de Queiroz, A matlab-based control systems laboratory experience for undergraduate students: Toward standardization and shared resources, IEEE Transactions on Education, 45(3), 218-226 (2002)

[11] C. Bonivento, L. Gentili, L. Marconi and L. Rappini, "A web based laboratory for control engineering education", Second International Workshop on Tele-Education in Engineering Using Virtual Laboratories (2002)

[12] J.L. Diez, M. Valles, A. Valera and J.L. Navarro, "Remote industrial process control with Marlab web Server", Internet Based Control Education (2002)

[13] J. Overstreet and A. Tzes, "An Internet based real-time control engineering laboratory", IEEE Control Systems Magazine, 9, 19-34 (1999)