

LABNET: A Remote Control Engineering Laboratory

Nourdine Aliane, Aída Martínez, Alberto Fraile and Jaime Ortiz

Universidad Europea de Madrid, DACA, Madrid, 28671 (Spain)

Abstract—This paper presents LABNET, an internet-based remote laboratory for control engineering education developed at UEM-University. At present, the remote lab integrates three basic physical systems (level control, temperature control and ship stabilizing system). In this paper, the LABNET architecture is presented and discussed in detail. Issues concerned with local or remote feedback, automatic report generating and reusing of experiment templates have been addressed. Furthermore, the experience gained developing, testing and using the system, are also presented.

Index Terms— Remote Laboratory, Control Engineering Education, Internet Based Control.

I. INTRODUCTION

The Internet becomes increasingly important as a learning environment, and internet-based technology is rapidly being adopted in engineering education as a tool for enhancing the educational experience enabling access to remote labs and performing real experiments, either as part of an on-site or distance learning course.

Remote laboratories are real equipments that can be operated and controlled remotely through an experiment interface [1]. They makes available a wide range of lab resources and can provide for more efficient sharing of unique or expensive, and simplifies the logistics and requirements involved in conventional laboratory work like scheduling of equipment, lab space, staffing and safety. Students can use the remote lab at any time and at any place to realize more experiments.

Technological development in remote laboratories shows significant progress during last years. In this context, numerous projects investigating remote access to laboratories used in science and engineering education are funded by the European Union. The most representative includes CYBERLAB [2], where it is proposed to form a network of providers of remote laboratories and to assign the technical, managerial and financial tasks involved in operating such a network to a commercial organizing company. PEARL [3] is another project which proposes to develop a system enabling students to conduct remote experiments over the WWW and research the pedagogic impact of this approach validating its developments in different educational contexts. Another EU project, known as MARVEL [4], proposes to develop an e-learning environment as a ‘mix’ of virtual labs in combination with remote experimentation.

In the control engineering area, the idea of sharing laboratory equipment using Internet for education purposes is not new. In the nineties, a remotely shared

control system laboratory had been proposed in [5], which enables sharing of laboratory data between universities, and the earlier implementation is reported in [6]. Since then, a variety of remote lab systems have been developed. Some realizations of remote labs include the web laboratories of the University of Tennessee at Chattanooga [7], the remote laboratory of the Stevens Institute of Technology [8] and the automatic control TELELAB of the University of Siena [9], [10] and [11]. All these remote labs offer online experiments in process control systems, allowing users to control a number of systems such as level control, temperature control and position/velocity control. Finally, the reference [12] gives a good review of current remote experimentation projects from academic and non-academic institutions, and pedagogic issues of online labs are discussed in [13] and [14].

This paper presents LABNET, an internet-based control engineering laboratory developed at the UEM–University. It enables users to easily and rapidly test a wide range of experiments in classical control engineering. It is intended to be transparent to end users, and it is also designed to facilitate the common administrative tasks. Currently, it is used for teaching “Control Engineering” subject to electrical and computer science undergraduate students.

The remainder of this paper is organized as follows: Section 2 describes the LABNET architecture and gives the main design goals. The client application used to access the remote lab from the user’s perspective is presented in section 3 and the server from the administrative point of view is presented in section 4. Considerations about the implementation are given in section 5. Section 6 describes the student evaluation, while conclusions and future works are drawn in section 7.

II. SYSTEM ARCHITECTURE AND GOALS

The LABNET framework is based on client/server architecture as depicted in Fig.1 where the communication is based on the standard Ethernet network with TCP/IP as protocol. The computer server is connected to real systems via two data acquisition cards and, at present, the remote lab can integrate four physical systems. The experiments concerned within the LABNET framework use open or closed loop with fixed structure controllers which can be executed on the server side or in the client side [15].

The LABNET software is organized in two internet-based applications; the lab-client and the lab-server. The lab-client application provides a virtual access to the laboratory through a GUI to perform experiments on real equipments. In the server side, the lab-server is the unique application used to perform administrative tasks such as

user management, connecting models and assigning their input/output, and managing with the remote experiments.

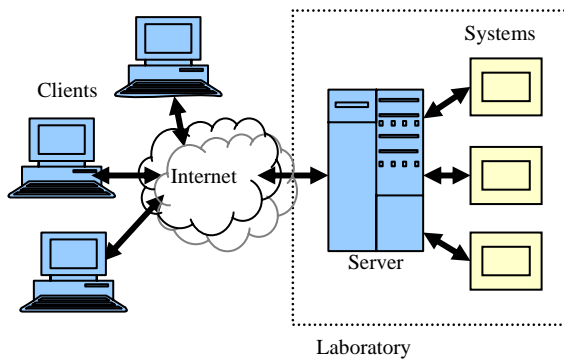


Figure 1. System architecture

A. Local or remote control

The LABNET permits the users designing a closed loop system with two configurations: local control and remote control

In the local control option, as illustrated in Fig. 2a, the controller is running in the server side. In this case, the user specifies all the parameters that define the experiment by interacting with the client and then submits the experiment specification to the server. There is no user interaction during the experiment. When the experiment is dispatched and completed, all the data such as system's response, control signal and the input reference are packaged and sent back to the client.

In the remote control option, the controller and the plant are physically located at different sites and are linked by the network as illustrated in Fig. 2b. This configuration is known as direct structure in the overall networked control system (NCS) [16]. In order to perform remote closed-loop control, the control signal is encapsulated in a packet and sent to the plant via the network. The server then returns the plant output to the controller by putting the measured data into packets as well. In this case, the performance of the closed loop is always affected by network delays since the network is tied with the control system.

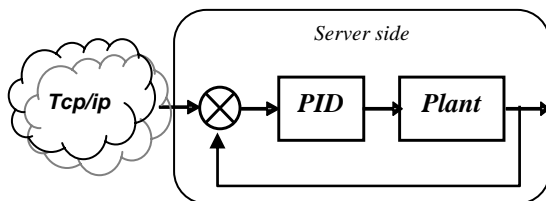


Figure 2.a. Local control

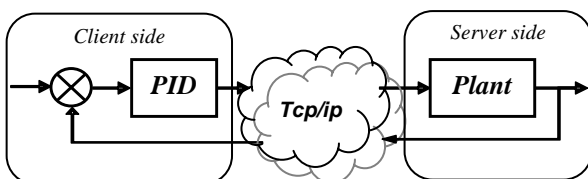


Figure 2.b. Remote control

B. The controlled systems

At present, LABNET integrates three types of models widely used in control engineering laboratories: level control, temperature control and ship stabilizer system.

The level control system consists of a liquid closed system with two measuring transducers of flow and level. Its control purpose is to maintain the desired level of the liquid in the tank by changing the inflow, while the outflow represents the disturbance in the system.

The temperature control system is a plant completely simulated by means of analogue circuits. The control goal is to maintain the desired temperature while changing the external temperature or opening the window.

The ship stabilizing system is a model with two ballast containers of closed liquid system with a position and velocity feedback variables. The system integrates a sea-wave simulator as system disturbances. The control goal is to maintain the desired inclination, while changing the load or activating the sea-wave.

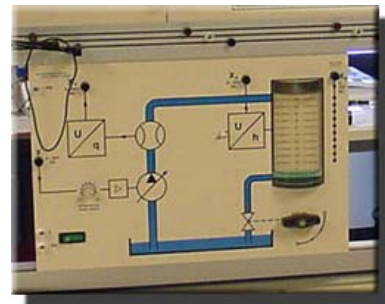


Figure 3.a. Level control system

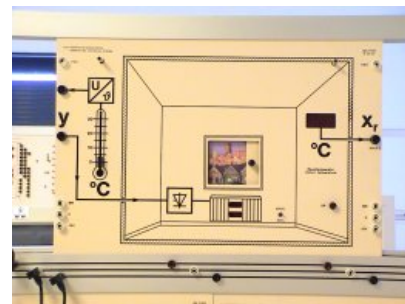


Figure 3.b. Temperature control system

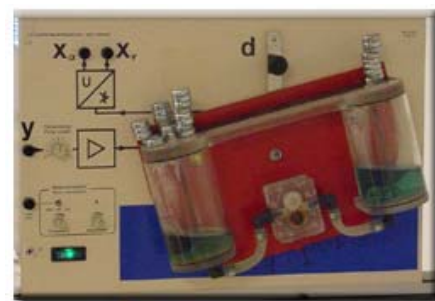


Figure 3.c Ship stabilizing system

All these models, are installed in their experimental panel, do not need any self-protection device to prevent damages or break the experiment availability.

C. The controllers

The PID controller is a proved solution for the most of industrial control applications. Its development went from pneumatic through analogue to digital controllers, but the basic actions remain the same. To obtain a good controller, it is also necessary to consider other aspects like the noise filtering, the windup phenomena, digital implementation, and so on [17]. All these issues are handled through the implementation of the following structures:

- A theoretical PID
- PID with 2-DOF and filtered derivative
- PID with an Integral Anti-Windup

The discrete controllers are obtained from the continuous ones using the Tustin's approximation. To prevent the control signal from wrong setting, software saturations have been added according to the model used in the experiment. Finally, digital implementations are validated using comparative tests with Matlab/Simulink simulations.

D. LABNET Capabilities

The LABNET environment enables users to easily and quickly test a wide range of experiments in classical control engineering. Some of these experiments are:

- Characteristics of open and closed loop system.
- Differences between P, PI or PID feedback.
- Effect of varying the tuning parameters on the control performance.
- Practical issues in control using a 2-dof structure and the PID with anti-windup.
- The effects of the sampling time.
- The effect of the delay in the remote control.

In addition, all the models integrated in LABNET exhibit substantial differences in their dynamics, so different approaches in the controller design are possible.

III. THE LAB-CLIENT

The lab-client is an internet based application that provides a virtual interface to the lab equipments by means of a GUI, as shown in Fig. 4, through which users can submit experiments specification and manipulate results. The interface is aimed to be simple to use and the user can focus his efforts on interacting with the experiment without wasting time. Every user has to enter a valid user name and password to access the lab-server.

Besides the experiment specification, lab-client provides additional features such as data visualization, automatic report generation and template manager.

A. Experiment specification

The user is able to personalize the experiments with different options. Firstly, the user chooses the model to experiment with, selects the local or remote configuration, and selects a predefined controller for the process and sets up its different parameters. Many others data such as sample time, the experiment's time or choosing the input reference must be also set up before running the experiment. The user has another possibility to specify his experiment by loading a stored experiment template. To prevent users from wrong setting in the different parameters, the lab-client provides a predefined signal generator with all parameters (sample time, amplitude and frequency) controlled to be within a predefined range.

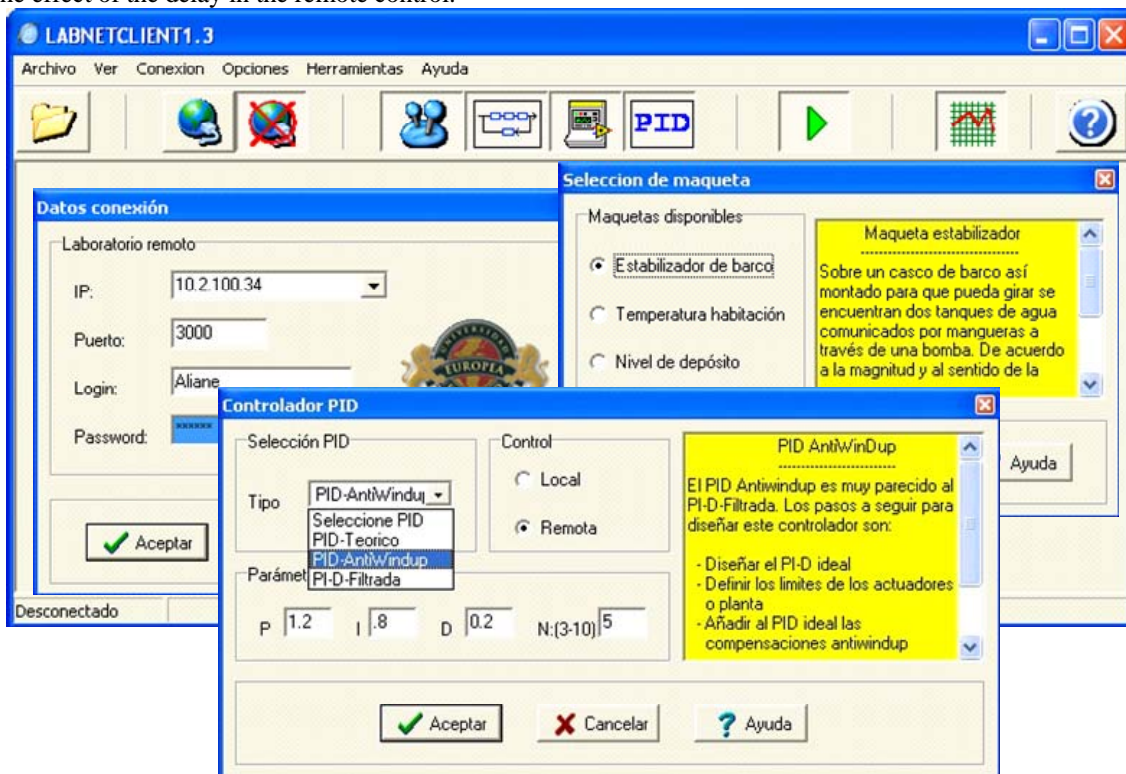


Figure 4. Lab-Client GUI and some dialogue boxes for experiments specification

B. Data visualization

All the signals involved in the remote experimentation, such as input reference, the systems' response or the control signal can be displayed with respect to time. The graphical scope, as shown in Fig. 5, allows the user to adjust the range of input values and automatically resized during the experiments. When an experiment is completed, the user can export and save data in many formats, such as HTML, Excel and ASCII data file, for future analysis and manipulation. The graphical results can also be saved in a GIF, BMP or JPEG format.

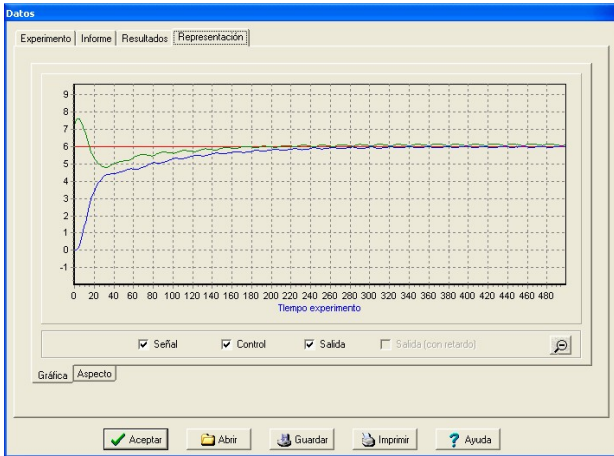


Figure 5. Visualization of the system's response

C. Report generation

Once the experiment is completed and all data are sent back from the server to the client, the user can generate automatically a report of his experiment. It is a structured file that contains information such as user identification, the controlled system used, the controller and its setting parameters, and so on. The report includes some links to the different data files and graphical used in the experiment. Students can upload and send the reports of their experiments to an e-mail address for the submission of their work. Fig. 6 shows an example of a report generated by lab-client.

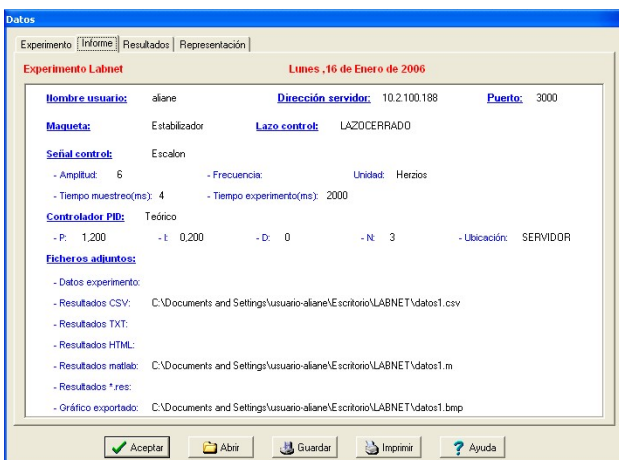


Figure 6. Model of report of an experiment

D. Template manager

LABNET provides users with a template facility for arranging experiments. Once an experiment has been completed, it can be saved as a template and can be reused later. Opening an experiment template brings the user all the settings and can run it immediately. The user can also make some changes in the experiment specification and does not need to repeat completely the setting process. An experiment template is a structured file with all the parameters used in an experiment specification. At beginning, all the experiments are initialized by a default template. Finally, experienced user can specify his experiment writing directly in the template file.

IV. THE LAB-SERVER

LABNET server computer is connected to real systems via two Advantech PC-LAB-816 DAQ cards for ISA-bus, where each card contains two analog output channels. Hence, the server enables us to make four single closed loops. The lab-server application provides a GUI, as depicted in the Fig. 7, through which the administrator can access all the administrative tasks. Lab-server has two different modes of operation: off-line and on-line mode.

- In the off-line mode, the lab-server is normally stopped and does not allow any remote connections and is limited to carry out administrative tasks such as user management, connecting models and assigning their input/output channels.
- In the on-line mode, the lab-server is started and listening to a specified port, waiting for a client to request a connection and enables a remote experimentation.

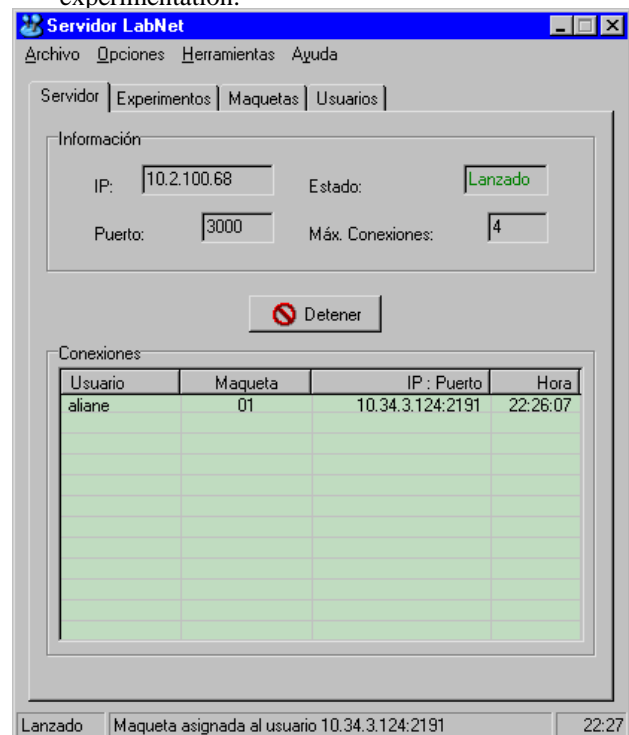


Figure 7. lab-server GUI

Referring to security aspects, LABNET is designed to be password protected and allows access only to registered users. The authentication is based on user name and a password. Furthermore, lab-server application is protected behind a firewall, where the administrator specifies the applications and ports that are allowed to receive incoming traffics.

A. Systems and input/output assignment

Within the LABNET framework, the real systems are handled through their corresponding virtual devices, which maps to the real systems. A virtual device is a software object with predefined polarities and full scale of the voltage signals used by its corresponding real system. The ultimate goal of the virtual device is to minimize the administrative tasks performing an easy input output assignments to the controlled systems. A virtual device (ultimately, a system) can be assigned arbitrarily an input and output from the DAQ cards as shown in Fig. 8.

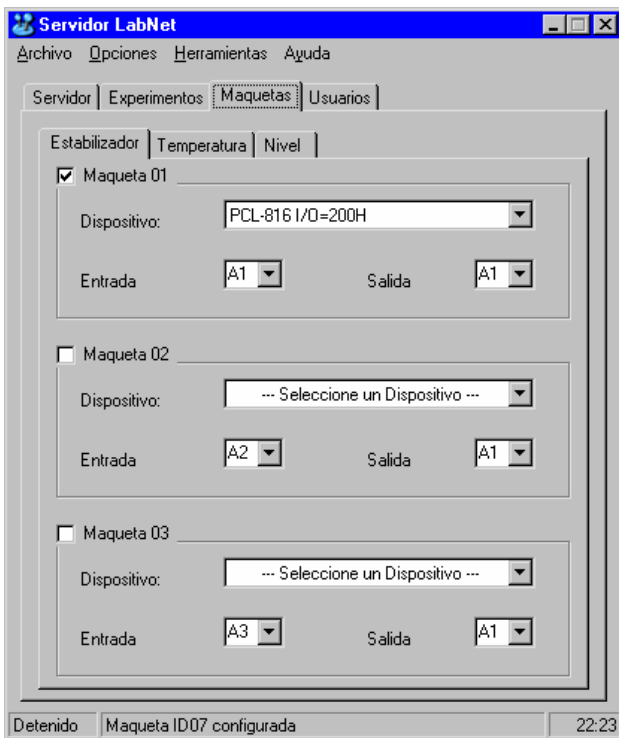


Figure 8. Input/Output assignment

B. Experiment status

All the administrative information maintained by the server can be retrieved and monitored in real time. This information concerns only the running experiments. It gives mainly the user identification, the models and the controllers that are being used together with many other parameters. The typical information displayed in the experiment status box is given in Fig. 9.

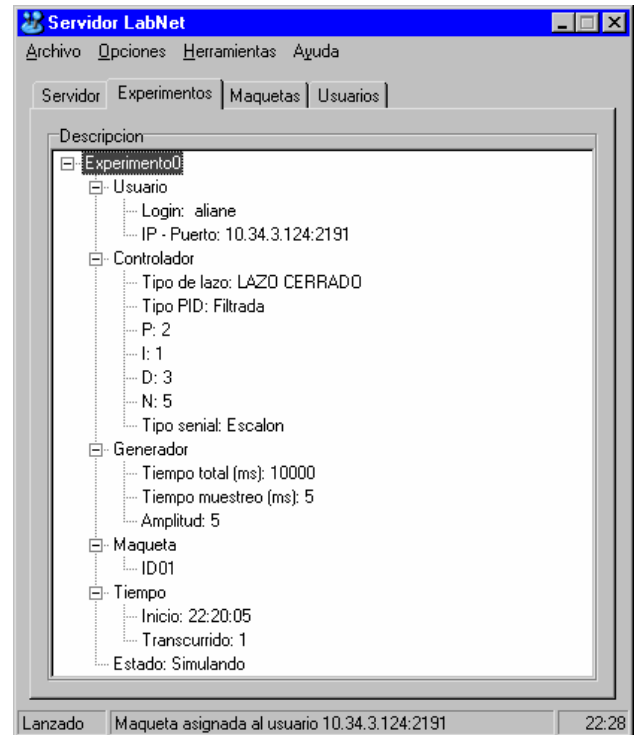


Figure 9. Experiment Status

V. SOFTWARE CONSIDERATIONS

Both lab-client and lab-server are implemented with an object oriented design to incorporate modularity and flexibility. Moreover, lab-server is designed as a multithreaded application and is able to serve up to four users simultaneously. The concurrency programming uses thread-per-session policy. ***In this model, each client that connects with a lab-server is served in a separate thread for the duration of the session and the same thread carry out all the requests received from this particular client. This model is suitable for concurrent remote lab servers that carry on long-duration conversations with multiple clients.

The synchronization and the sampling time are driven by the windows multimedia timer [18] which allows high-resolution timing. A sample time of 4ms with good robustness is reached using a server as Pentium-III under windows 2000. The unpredictable behavior takes places when the resolution is less than 4 ms.

As the PCL-816's DLL driver works seamlessly with C/C++ programming language, C++ Builder has been chosen as a development tool.

VI. STUDENTS EVALUATION

To evaluate the convenience of LABNET, we offered our students an option to repeat part of the experiment they had performed in regular laboratory classes to use it as a self-training tool. The remote experiments consist in exploring aspects of controller tuning. The task required them to set the gains of the controllers and then measure performance parameters such as rise time, overshoot, static error and the integral of square error performance index. At the end, they were also asked to answer a small questionnaire.

Using LABNET, 68% of the students (17 of 25) managed to operate the remote lab for more than 30 minutes operating the equipments for themselves, and about 25% operated the remote lab for more than 60 minutes. All the connections were from the campus and around 55% of these sessions were “in situ” from the control laboratory and in presence of a teacher.

The attitude of the students towards working with this new tool was positive and the majority reported that it was easy to use. Most of them declared that the LABNET has extended their lab experience. The major point of criticisms was pointed around the lack of sense of presence during experimentation.

At present, LABNET is used during regular lab sessions and used during lectures for demonstration from the class-room where normally no equipments are present using wireless connection.

VII. CONCLUSION AND FUTURE DIRECTION

LABNET, an internet-based remote laboratory for control engineering education developed at the UEM-University, has been presented. It enables users to easily and rapidly test a wide range of experiments in classical control engineering.

The remote user only requires the lab-client, an executable program which can be downloaded from the laboratory web page. It runs in all MS-Windows platforms. On the server side, lab-server is the unique program required to perform both administrative tasks and the managing with the remote experiments.

The LABNET framework provides important features like a concurrent access up to four users, local or remote feedback, automatic report generating, and reusing experiment templates.

Current version is considered the first prototype, and a range of actions are planned to perform substantial improvements centered in functionality and experimental capability. To improve the LABNET functionality, one of our next objectives is its integration in the MOODLE e-Learning platform. This permits us to carry out an integral course management and to inheritance many embedded features such as the booking mechanism, associating eLearning contents with remote experiments, and many other utilities that provide an excellent match to the pedagogical requirements of remote experiments. Other improvements will be focused on the addition of new utilities to extend the LABNET experimental capability such as identification systems, auto-tuning utilities, benchmarking facility to store and compare experiments and to adapt LABNET as multilingual application.

REFERENCES

[1] Dormido, S. “Control learning: Present and future”, Proceedings of the IFAC 15th World Congress, Barcelona-Spain, 2002

[2] CYBERLAB: http://www.cyberlab.org/frames_cl.php

[3] Cooper, M. “PEARL: Effective learning through remote experiments”. A collection of papers Retrieved September 26, 2005 from: <http://iet.open.ac.uk/pearl/index.htm>

[4] Müller D, “MARVEL – Mechatronics Training in Real and Virtual Environments”: A collection of papers Retrieved, 2005 from: <http://www.marvel.uni-bremen.de/index.php?id=72>

[5] Aburdene, M et al, “A proposal for a remotely shared control systems laboratory”, Proceedings of the ASEE, Frontiers in Education Conference, pp. 589-592, 1991.

[6] Aktan, B and C.A. Bohus, “Distance Learning Applied to Control Engineering Laboratories”, IEEE Transactions on Education”, vol 39 (3), pp 320-326. 1996

[7] University of Tennessee at Chattanooga, Web Laboratories, <http://chem.engr.utc.edu/>

[8] Stevens Institute of Technology, Remote Laboratory, http://www.soe.stevens-tech.edu/Academics/remote_lab.html

[9] Casini, M y D. Prattichizzo, “The Automatic Control Telelab: A User-Friendly Interface for Distance Learning”, IEEE Transaction on Education, 46(2), (2003)

[10] Casini, M, D. Prattichizzo y A. Vicino, “The Automatic Control Telelab: A Web-based Tecnology for Distance Learning”, IEEE Control Systems Magazine, 24(3):36-44, (2004)

[11] University of Siena, Automatic Control TELELAB <http://www.dii.unisi.it/~control/act/home.php>

[12] Callaghan M. J et al., “Client-Server Architecture for Remote Experimentation for Embedded Systems”, International Journal of Online Engineering, 2 (1), (2005).

[13] Müller, D. y J.M. Ferreira, “Online labs and the MARVEL experience” International Journal of Online Engineering, 1(1), (2005).

[14] Cooper, M. “Remote laboratories in teaching and learning – issues impinging on widespread adoption in science and engineering education” International Journal of Online Engineering, Vol. 1 (1), (2005).

[15] Overstreet J.W. and Tzes, A “Real-Time Control Engineering Laboratory”. IEEE Control Systems Magazine, October, pp 19-34, 1999.

[16] Yodyium Tipsuwan and Mo-Yuen Chow “Control methodologies in networked control systems” Control Engineering Practice Vol 11, pp 1099–1111, (2003)

[17] Åström, K, J and B. Wittenmark: Computer Controlled Systems. 3ed Edition. Prentice Hall, (1999)

[18] <http://msdn.microsoft.com/library/>

AUTHORS

Nouridine Aliane is with the Computer Architecture and Automation Department (DACA) at the Universidad Europea de Madrid -Villaviciosa de Odón, Madrid, Spain 28671- e-mail: aliane@uem.es

Aída Martínez, Alberto Fraile and Jaime Ortiz are Graduate Students in Computer Science at UEM-University

Manuscript received November 17, 2006.