MPTL14 2009
Udine 23-27 September 2009

THE FISL@BS PORTAL: A NETWORK OF VIRTUAL AND REMOTE LABORATORIES FOR PHYSICS EDUCATION

Luis de la Torre, José Sánchez, Sebastián Dormido, *Informática y Automática Department, Spanish University of Distance Education (UNED)*

Abstract
The FisL@bs portal constitutes a distributed network of remote and virtual laboratories for science physics education via Internet. FisL@bs offers students the possibility of performing hands-on experiences in different fields of physics in two ways: simulation and real remote operation. This paper gives an overview about how this portal works and the hardware and software tools used to create it.
One of the experiments already operative is the one used to discover and verify Hooke's law. The simulated experience is briefly described and a detailed description of the real assembly and the remote control and experience is given. As every other laboratory at FisL@bs, it is accessible for anyone with an Internet connection and a Java compatible web browser.

1. Introduction
Web-based -simulated and remote- laboratories are quite well established in several scientific and technical disciplines. While simulation serves important purposes, real laboratories can not be substituted just with this tool, especially in some fields such as physics, in which the actual behavior and response of the real elements in the experiments is crucial (Alhalabi 2000). Although remote laboratories -which deliver laboratory facilities to the door of the student such as the simulation software is intended to do (Gorrel 1992)- may constitute a better substitute or complement to real hands-on experiments, virtual laboratories may still be used to serve as an initial experimentation and a first contact with the studied phenomena for the student.
FisL@bs is a distributed network of both remote and virtual science physics laboratories for higher education via Internet. This network will be distributed among different Spanish universities and uses the same structure as AutomatL@bs (Vargas 2009), a network of virtual and remote laboratories for learning/teaching of control engineering, which is operative from 3 years offering experiments such as the three-tank system (Dormido 2008). FisL@bs offers students the possibility to realize hands-on experiences in different fields of physics in both a real but remote way and in a simulated way. The simply way by which students can experiment with these laboratories -using a Java compatible web browser with and internet connection- is easy and common enough to grant access for all of them.
As part of the FisL@bs project, several laboratories are already available or still under development: a motorized optical bench for the determination of the focal length of a thin lens, a Snell's law reflection/refraction experiment, a rigid pendulum, a Hooke's law experiment or a sensor whose XY position can be remotely controlled -quite similar to a plotter- to measure the distribution of potential over a resistive sheet of paper with different electrostatic field configurations. As said before, all these laboratories are all accessible for anyone with an Internet connection and a Java compatible web browser.
In order to reduce costs, make it easy to build and give a user friendly impression to students, some of the previous experiments -such as Hooke's law experiment and the plotter- were built using Lego Mindstorms pieces. The intelligent NXT brick from Lego lets a computer -via USB or Bluetooth- to control the motors and to read the sensors connected to this brick. The real-time software that carries out these control and measurement tasks was programmed with LabView (Travis 2000), a graphical programming language specifically designed for developing instrumentation, diagnostics, and data acquisition systems. The graphic user interface (GUI) to remotely experiment with this laboratory is a Java interface created with Easy Java Simulations (EJS), an authoring tool written in Java that helps to create interactive simulations in Java, mainly for teaching and learning purposes (Esquembre 2004), (Christian 2007). The virtual experiment has also been made using EJS. Finally, eMersion (Gillet 2005), a Web-based learning environment which facilitates the deployment of pedagogical scenarios and learning resources for Web-based experimentation in education, is also used to hold the developed EJS interfaces.
In order to provide communication between Java programs and Labview virtual instruments, a web-enabled environment for remote diagnostics, maintenance, and experimentation in

engineering which is based on a middleware layer that uses a Java-Internet-Labview server must also be used. This environment -consisting on a stand-alone application called JIL server (Vargas 2009) and a Java library file- uses Labview on the server side (where the real experiment is located), Java applets on the remote client side (the student's computer) and TCP/IP as the communication mechanism between both elements.
The rest of the paper is organized as follows: section 2 offers a brief overview of the structure used by FisL@bs, section 3 describes one of the particular experiments -Hooke's law- available in this network of remote and simulated laboratories, and section 4 contains our conclusions.

2. Web-Based Laboratories Environment

Following the web-based laboratories structure used for the AutomatL@bs project (Dormido 2008), FisL@bs uses a client-server architecture, where TCP/IP is the protocol of communication used to carry out the data exchange between them. Fig. 1 shows this communication architecture.

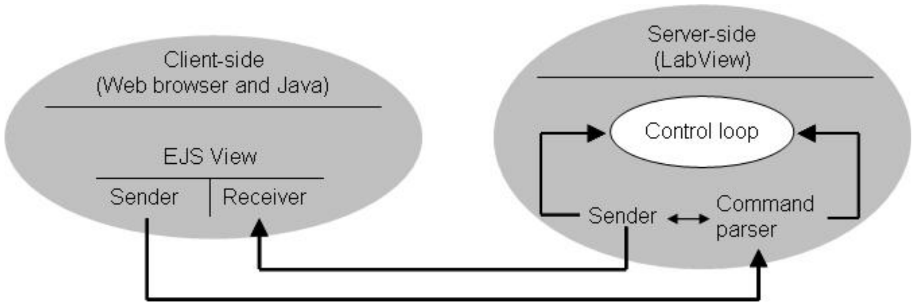

Figure 1: Client-server architecture

2.1. Client-side implementation

The client side is a Java applet created with EJS and contained in the eMersion environment. When connected to a laboratory at FisL@bs, any common web browser will show 3 windows (see Fig. 2). The upper window lets users to: 1) examine the list of activities he must execute, 2) examine the provided documentation, 3) change the language, and 4) log off. The right window shows the files generated by the use of the laboratory such as data files and graphics images. Finally, the biggest window contains the applet with the experiment. Initially, this applet shows the simulated experiment which gives both the visualization and the interaction needed to control it.

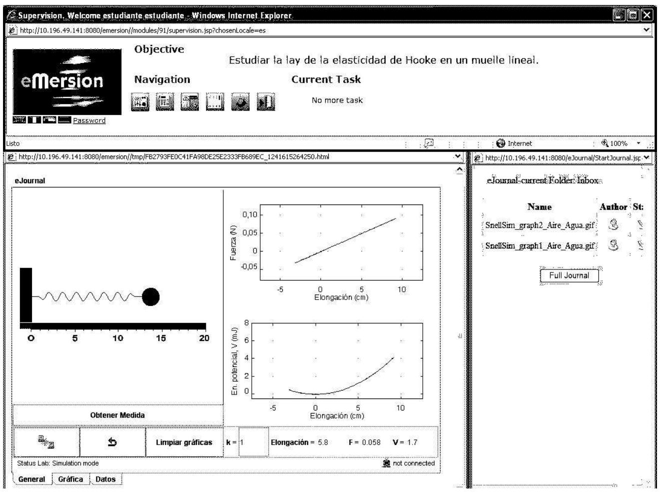

Figure 2: eMersion and Hooke's law simulated experiment

Although the simulated experience runs on the client side, once the student has successfully dealt with the virtual experiment, he would be allowed to take the next step: the access to the remote laboratory. The Java applet made with EJS contains a button to connect to the real laboratory and, after pressing this button, the applet changes from the view of the virtual experiment to the real one. In this client side, interaction with this laboratory is also done by means of this applet -the EJS view-, which counts with special calls (the sender and receiver commands in the client side of Fig. 1 in order to establish communication with Labview by using TCP functions and routines. While the receiver command is continuously used, the sender command is only executed when the student changes a value in the EJS view.

2.2. Server-side implementation

The same computer acts as Web server as well as the controller. This controller is developed in Labview and it contains two main information loops. The first one is an asynchronous loop that is in charge of communicating with the EJS view (receiving the users' actions and sending the system state information) while the second one is a synchronous loop that continuously controls the experiment.
The first -asynchronous- loop may present some problems when trying to control or experiment with certain processes due to its fast dynamics and the possible network delays. However, this is something that must be analyzed for each particular experiment and, if any problem appears, there is not a unique way to deal with it.

3. A Mechanics Experiment: Hooke's Law

In mechanics, Hooke's law of elasticity states that the extension of a spring is in direct proportion with the force applied. This statement is true as long as this force does not exceed the elastic limit. Mathematically:

$$F(t) = - k \cdot (x(t)-l) \quad (1)$$

Where k is the spring constant, l is the natural length of the spring, $x(t)$ is the actual length of the spring at time t and $F(t)$ is restoring force exerted by the material at that moment.
Hooke's law real experiment uses the next elements: one NXT intelligent brick, a spring with constant k=14 N/m, one NXT dc motor, a 25 cm long rack rail, one contact sensor, a Vernier force sensor and a ruler. The spring has one of its extremes fixed to a wall, while the dc NXT motor -which is mounted over the rack rail- is attached to the free extreme of the spring so the motor can stretch the spring when it moves along the rack rail. The contact sensor is used for resetting purposes when the motor reaches the end of the rack rail and the force sensor used has a wide enough measurement range ([-10 N, +10 N]) and a good enough precision (±0.01 N). Finally, the ruler (not present at the moment of taking the picture of Fig. 3) has markings with 1 mm graduations. Fig. 3 shows the actual assembly of this experiment.

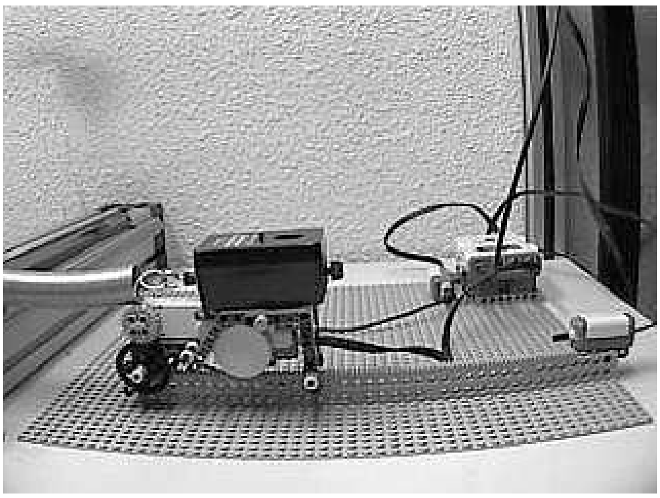

Figure 3: Hooke's law experiment assembly

In this experiment, students can control the position of the NXT motor over the rack rail and thus, the actual length of the spring. The NXT motor has a rotation encoder with ±1º precision which translates into about ±0.4 mm precision for the spring stretching -considering that the error of the

known value, l, is zero-. Thus, students control the x variable in equation (1) with ±0.4 mm of precision. However, students may also use the ruler in order to measure the spring stretching and so the precision would be ±1 mm. F is the variable measured by the force sensor and given to students with ±0.01 N of precision. The natural length of the spring, l, is known -its associated error is not taken into account- and the spring constant, k, is the unknown. Therefore, in this case there is just one control parameter sent from the EJS view to the hardware controller: the position of the NXT motor over the rack rail (x). The returned information is the measured force and the position of the motor encoder translated into the actual position of the NXT motor.
Since the cross-sectional area of the spring, A, is also known -and given to the students- it is even possible to obtain the modulus of elasticity, E, using equation 2 once the value of k has been determined:

$$k = {}^{A \cdot E}\!/_{l} \tag{2}$$

3.1. Simulation activities

The simulated experiment lets students to discover Hooke's law and to see how the potential energy (V) of a spring changes when it is stretched or compressed. In Fig. 2, the upper graphic shows Hooke's law linearity while the other one shows V against the string stretching/compression (x-l). These graphics change dynamically when the student moves the ball attached to the spring -which is done dragging it with the mouse- in order to either stretch it or compress it. A default value for the spring constant (k=1) is used in the simulation but the user may change it just writing the desired value in the numeric input field.
Data and graphics can be saved and so, students can compare the theoretical (simulated) results with the real ones (see next section) in order to analyze the existing differences.

3.2. Remote activities

When connected to the remote laboratory, students will be able to check Hooke's law using a real experiment. Fig. 4 shows the GUI in this mode. Two displays are shown: the first one gives the force measured by the sensor while the second one shows the spring stretching. The slider bar is used by students to change the position of the motor over the rack rail and thus, the spring stretching. On the right side, a webcam shows the experiment in live while the graphic on the left side changes dynamically, plotting the force against the stretching.

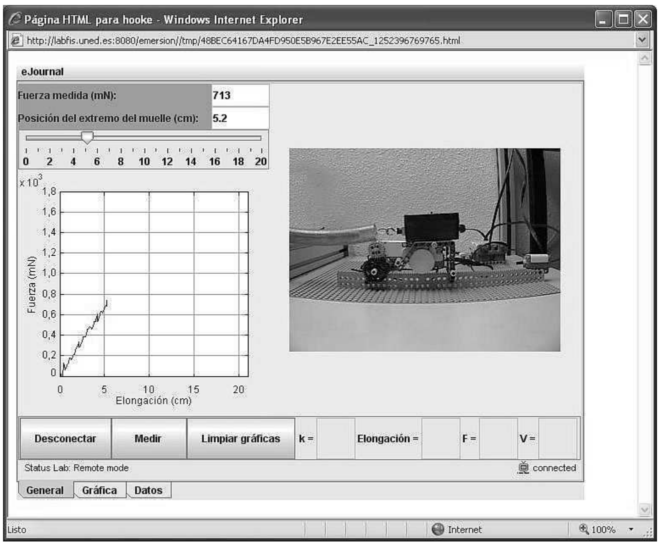

Figure 4: Hooke's law remote experiment

The physics phenomena may be quite simple in this experiment but there are still so many other important activities to carry out and lessons to learn related with metrology by means of this laboratory.

For example, measurement uncertainties must be taken into account by students. As said before, the measurement uncertainties in this experiment are: $\delta x = \{\pm 0.4 \text{ mm}, \pm 1 \text{ mm}\}$ (depending on how the spring stretching is measured: using the motor encoder or the ruler), $\delta l \approx 0$ and $\delta F = \pm 0.01$ N. Considering the last value and the elastic constant of the spring (14 N/m), students should realize that they must take measurements with at least 0.7 mm of difference between each stretching in order to be able to measure force differences. Also, uncertainties calculus can be practiced here since students should give their results -the spring constant and the modulus of elasticity values- with their appropriated uncertainties. Following these ideas, this experiment may also serve students to learn the importance of using well-balanced equipment, since a great precision -small uncertainty- for the stretching measurements (using the NXT motor encoder) is considerably wasted when determining the spring constant value because of a quite poorer precision for the force measurements. Finally, in order to obtain *k*, students are asked to do a linear regression with the collected data (*F* against *x-l*) for both the simulated and the remote experiment and to compare the results with the graphics shown in Figs. 2 and 4. Therefore, adjustment by least squares -manually and/or computerized- are also practiced by students.

4. Conclusions

FisL@bs continues the work initiated with the AutomatL@bs project, expanding its utility from the control engineering education to the science physics education. FisL@bs inherits the well-proven structure of its successful counterpart and just changes the simulated and remote experiments in order to make it suitable for a physics course.

The simulated experiment serves students as a first contact with the studied phenomena in which they can, for the presented example, discover Hooke's law and study the potential energy variation in a spring. Continuing with this same example, the remote experience lets students to verify Hooke's law with a real experiment. Although this is a quite simple experiment, students would not only learn about springs and Hooke's law but also about measurement uncertainties, uncertainties calculus and linear regressions.

Other experiments are already operative or still in development, such as the rigid pendulum, the optical bench, the Snell's law experiment, etc. All of these -with both the remote and the virtual experiences- will be integrated into the network of physics laboratories that constitutes FisL@bs and they all use the same structure and the same software tools as the example presented in this work.

References

Alhalabi B, Marcovitz D, Hamza K and Petrie M (2000) Remote Labs: An innovative Leap in the World of Distance Education, Proceedings of the 5th IFAC Symposium on Advances in Control Education, Gold Coast, Australia

Gorrel J (1992) Outcomes of Using Computer Simulations, Journal of Research on Computing in Educations, 24, 359-366

Vargas H, Sánchez J, Dormido S (2009) The Spanish University Network of Web-based Laboratories for Control Engineering Education: The AutomatL@bs Project, European Control Conference, Budapest, Hungary.

Dormido R, Vargas H, Duro N, Sánchez J, Dormido-Canto S, Farias G, Esquembre F and Dormido S (2008) Development of a Web-Based Control Laboratory for Automation Technicians: The Three-Tank System, IEEE Transactions on Education, 51(1), 35-43

Travis J (2000) Internet Applications in LabView, Prentice Hall

Esquembre F (2004) Easy Java Simulations: A Software Tool to Create Scientific Simulations in Java, Computer Physics Communications, 156(2), 199-204

Christian W and Esquembre F (2007) Modelling Physics with Easy Java Simulations, The Physics Teacher

Gillet D, Nguyen Ngoc A V and Rekik Y (2005) Collaborative Web-based Experimentation in Flexible Engineering Education, IEEE Transactions on Education, 48(4), 696-704

Vargas H, Sánchez J, Dormido S, Salzmann C, Gillet D and Esquembre F (2009) Web-Enabled Remote Scientific Environments, Computing in Science and Engineering, 11(3), 36-46

Dormido S, Vargas H, Sánchez J, Dormido R, Duro N, Dormido-Canto S and Morilla F (2008) Developing and Implementing Virtual and Remote Labs for Control Education: The UNED pilot experience, 17th IFAC World Congress