A Remote and Virtual Lab with Experiments for Secondary Education, Engineering and Lifelong Learning Courses

Alberto Cardoso, Miguel Vieira
1CISUC - Centre for Informatics and Systems of the University of Coimbra
Department of Informatics Engineering
Coimbra, Portugal
alberto@dei.uc.pt, mavieira@dei.uc.pt

Paulo Gil1,2
2Departamento de Engenharia Electrotécnica Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa, Campus de Caparica, Lisboa, Portugal
pgil@dei.uc.pt

Abstract— The Information and Communication Technology tools are nowadays fundamental to support e-learning and b-learning courses. The Remote and Virtual Laboratory available at the Department of Informatics Engineering of the University of Coimbra (Portugal), RVL@DEI-UC, is a web-based platform that allows users to perform experiments in different educational levels and contexts. This paper aims to describe its potential use in secondary education, engineering and lifelong learning courses. The conceptualization, architecture and implementation of the web platform for real and virtual experiments, remotely controlled using the Internet, are presented and the relevance of the lab’s integration in an intelligent tutoring system is also discussed, mainly in what concern the requirements of adaptation to users’ profile in different learning contexts.

Keywords- Remote and virtual lab; intelligent tutoring system, b-learning; secondary education; engineering education; lifelong learning.

I. INTRODUCTION

The development of remote and virtual laboratories (RVLs) have received considerable attention in the last few years, specially in building web-based educational tools [1], [2], [3] and platforms for students at secondary and higher education levels, as well as for lifelong learning courses. The relevance of the integration of RVLs in intelligent tutoring systems is also of great interest, especially to introduce the requirements of flexibility and adaptability in such systems [4].

This paradigm of learning and training promotes the development of web platforms rich in educational contents and the usage, in a flexible and easy way, of remote and virtual experimental setups, breaking down spatial and temporal barriers of traditional approaches. It just requires an Internet connection to access different kinds of virtual or real processes and experiments in an e- or b-learning context, which greatly contributes to the enrichment of classroom activities [5]. Students may accomplish remotely their own experiments from a laptop using an availability Internet connection. See e.g. [6] for some reflections and perspectives on the Systems and Control area of several Engineering courses.

As expected, several courses in different degrees are currently taking advantage of all the potential of Information Technologies, namely, the Internet infrastructure and web technologies, to develop and implement distant teaching and learning paradigms. In a web-based virtual laboratory students or trainees use the Internet or an Ethernet local area network (LAN) to perform experiments on setups expressed in the form of models of plants, being the virtual system’s time response emulated by simulation on a fast processor server [7, 8]. The outcomes of these simulations, i.e. the system time response and control actions, are subsequently retrieved and analyzed by users. In the case of remote laboratories, students, instead of performing experiments remotely on virtual plants, can have access to real pilot plants, connected via data acquisition boards to a server computer [9], [10], [11]. This kind of laboratories can operate either in batch mode or online.

The integration of the RVL in an intelligent tutoring system (ITS) represents an approach to build a platform with capability of adaptation to different users’ profiles. The ITS should be able to define contents and propose experiments according to the specific needs and backgrounds of each user. In particular, students from secondary education levels, engineering courses and trainees from industries have different profiles and the system should be developed taking into account their specific characteristics and needs. In general, the use of ITSs can provide advantages in the student’s and trainee’s exploration and interaction by creating adequate sequences of experiments and following up the user’s performance in real time, using, for example, task planning to enhance the user’s learning performance and providing feedback, help and guidance to address the user’s immediate problems and needs [12].

The Remote and Virtual Laboratory available at the Department of Informatics Engineering of the University of Coimbra (RVL@DEI-UC) is a laboratory with interactive computer applications that allows users to carry out remote and/or virtual experiments using real physical systems that can be remotely or locally accessed and controlled through a web browser. The experiments provided to each user through the intelligent tutoring system are selected according to its profile, background and the characteristics of the course [13].
Next sections outline the main characteristics of the RVL@DEI-UC, describe some aspects of the ITS and sketch some experiments that can be considered in secondary education, engineering and lifelong learning courses contexts.

II. THE REMOTE AND VIRTUAL LAB

The Remote and Virtual Lab (RVL@DEI-UC) is a hybrid laboratory with interactive computer applications that allows users to carry out real or virtual experiments on systems that can be remotely or locally controlled through a web browser using shared communication networks.

A. The RVL Conceptualization

The RVL@DEI-UC is a web based platform that enables accomplishing remote experiments in a real laboratory, using shared networks. For example, the user has the possibility to configure and observe the behavior, as well as the performance of local controllers, like in most systems of this nature. However, this approach offers to users, considerably more powerful tools, since users can also design their own controllers. This concept is different from classical approaches and falls under the category of a relatively new hybrid scheme where the controller's parameters are tuned on-line on the client side, while the server transmits the system’s response.

The advantages of this design are, among others, the ability for a client to use the platform to test controllers developed by him/herself, giving additional usability to the system as a resource for investigators; the ability to use the computational power of clients’ computers to implement more resource demanding control laws; the ability to use complex controllers despite significant limitations on the server side; the possibility of privacy for remote users who may not wish to send their controllers code for compilation and execution to the server; more flexibility in the controller design process, since the user has the freedom to implement any controller on the client side; better security against download viruses at the server side, since the server does not compile or run any remote software locally.

However, this paradigm introduces new issues to deal with, namely, communication failures and delays. For this reason this platform was designed to provide fault-tolerance mechanisms, which can address the problems derived from the unreliability of the Internet and the extra flexibility and responsibility that is presented to users. Additionally, the platform offers a wide range of monitoring features such as: numeric and graphical visualization of information related to the running experiment and the network communication; animated representation of the experiment, which can be shown in 3D, 2D or both; video and audio streaming of the experiment; the possibility to interact with a local operator, if available, through a microphone. Some of these features are uncommon in systems of this kind, namely the ability to access predefined performance metrics to analyze the quality of service of the network communication. Distinctly from most platforms any user has the possibility to use available monitoring features interchangeably or having them combined, for example, placing the animation on top of the video image in order to compare de accuracy of the virtual representation.

B. The RVL Architecture

The RVL@DEI-UC is based on a client-server architecture (simplified version in Figure 1) and consists of four main applications. The server application is running on a server computer at the laboratory site and the client application is installed on a remote computer. Additionally, the media server application runs in an additional computer, while the process application runs on each computer connected to a real process.

The server application establishes the interaction between either the real setup or the virtual process and the client application, located on a remote computer. It is implemented in JAVA and makes use of MATLAB functions, such as, for I/O communication with DAQ cards or to evaluate the sequence of control actions taking into account previous inputs and outputs, as well as reference trajectories. In case the platform is configured as a Virtual Laboratory the computed sequence of control actions are provided to the virtual plant environment, and used in the plant’s simulation model by calling the MATLAB computational engine.

Each real process is connected to an off-the-shelf PC equipped with input/output (I/O) interface boards from different manufacturers. The software used to interact with laboratory systems is based on Matlab/Simulink with the Humusoft Extended Real-Time Toolbox and considers safety measures to guarantee the setup integrity against misuses.

The client application is launched by a user logged in a remote computer, connected to the server via an Ethernet local area network or the Internet. After the authentication, the client application gets from the server a list of available real and virtual plants. For example, in a control experiment, the remote user should select and parameterize a given controller to run on the server, in case of batch operation mode or, alternatively, on the remote computer, in case of online control scheme. In this topology the computed sequence of control actions is sent to the server using a TCP connection. Users should also define other parameters, such as, the sampling time, the overall time for the experiment, the reference trajectories and the type of control cycle.

While the experiment is in progress, the remote user can follow the closed loop control performance in a dedicated window, displaying outputs, reference trajectories and the computed control actions. Additionally, the user may have access to a number of monitoring statistics regarding the running experiment. These statistics include, the updated number of timeouts, server-client and client-server latencies, just to name out a few.
C. The web based platform

To use the RVL, the user should log in the platform’s website and go through the requirements verification page to see whether he/she has the software required. Figure 2 shows the main webpage.

![Main web page of the platform.](image1)

After choosing the type of process, real or virtual, the website lists the available processes on the database and presents those ready to be used. For selecting and start running an experiment that is available (Figure 3), it is previously necessary to set the experiment’s configuration parameters.

![Example of experiment’s selection interface.](image2)

In the next step, the user could configure the experiment (Figure 4) which includes, for example, the control horizon, the sampling period and the contingency plan, in case there is a problem with the communication network [14].

![Connection configuration window.](image3)

Afterwards, users should define and configure the process controller by selecting and initializing a given controller. Figure 5 shows the controller window. Finally they are allowed to start running the experiment.

![Controller configuration window.](image4)

During the experiment users can follow the closed loop control performance in a numeric window and in a graphical window that displays outputs, reference trajectories and the computed control actions.

![Platform’s interface for visualization of an experiment concerning the control of a three-tank system.](image5)

Additionally, they may have access to a number of monitoring statistics regarding the running time, as well as, monitoring features: 2D/3D model, video and audio streaming. Figure 6 shows an example of the described monitoring features applied to a three-tank system experiment, including real-time graphic representation of the system’s inputs and outputs audio, video and 3D dynamic visualization.
The platform’s interface from the user’s point of view includes the following means of visualization:

- Numeric visualization of signals associated with the process and of network communication control metrics;
- Graphical visualization of process’s inputs and outputs and network communication control metrics;
- Dynamic virtual representation of the experiment and related values;
- Video and audio streaming.

The numeric visualization of signals associated with the process shows the inputs and outputs values of the process in real-time and is complemented by the graphical visualization of these variables, which allows the observation and analysis of the entire experiment progress. The numeric and graphical visualization of network communication performance metrics (Figure 7) is not commonly available at the client side in this kind of platforms. However, it is an important aspect that allows users to evaluate the network communication’s performance, validating the quality of service, detecting possible bottlenecks along the control cycle and taking advantage of acquired data to fine-tune his/her controller to adapt to changes in the network conditions.

![Figure 7. Platform’s interface for visualization of network communication performance metrics.](image)

The virtual representation of the system and its related variables and parameters is quite significant in virtual experiments, where video streaming is not present, simulating dynamically the evolution of the system’s variables and events taking place in a 3D environment. Nevertheless, this representation can also be accessed in remote experiments interacting with real processes and can be combined with the live video feed. Additionally, it is possible to overlap these two forms of visualization so the user can better assess the accuracy of the virtual representation.

The audio and the live video feeds are invaluable for making the remote experience as close to an actual on-site experience as possible. These elements along with the remote control over the experiment and the ability to extract data results are the main components and the most important tools offered by the present remote and virtual laboratory.

III. THE INTELLIGENT TUTORING SYSTEM

The interaction with the laboratory plants could be defined in function of a particular user’s profile or preconfigured according to a scheduled learning process. For instance, depending on the user’s skills, the control system can be established using a remote controller designed and implemented by the user on the client side, or considering a local controller with a predefined structure, but in which some of the controller parameters can be configured remotely. The set of experiments that can be performed through the RVL includes various processes with different levels of complexity that are chosen in an adaptive way, in order to fulfill the objectives and the characteristics of each learning program.

The integration of remote and virtual laboratories in intelligent tutoring systems is of great relevance for the learning process because users can have the opportunity to complement the theory and practice, which considers usually only simulators, developing and implementing control systems to interact with real or virtual laboratory processes. Moreover, building courses with requirements of adaptability to different profiles of students/trainees (K-12 students, engineering students or trainees in lifelong learning) can improve greatly the learning process, by motivating users and giving them conditions to knowledge acquisition with a priori good results.

The tutoring system under development at DEI-UC seeks to incorporate intelligent agents that implement mechanisms of identification, monitoring and supervision of users’ behavior and progress, namely, in terms of performance and accomplishment of tasks proposed in each course’s module. The ITS is supported on a Moodle platform and is being developed using the freeware CourseLab authoring tool. Figures 8 and 9 present examples of slides where users can obtain information about the experiment and interact with the remote and virtual lab and run an experiment, for example, to control a three-tank system in the context of an engineering course.

![Figure 8. Example of the introduction to an Experiment.](image)
Whenever the ITS identifies the necessity to make suggestions, warnings or indications to a user, an intelligent agent shows a virtual teacher that interacts or writes a message on the slide. The methodology and intelligence of agents are in process of research, having in mind that the quality of the system depends deeply on the agents’ ability to interpret the behavior and progress of each user and to react accordingly.

IV. EXPERIMENTS

Several real-time and virtual experiments can be carried out using the RVL@DEI-UC platform. Additionally, it enables to interact with different laboratory systems presenting dynamic characteristics compatible with common control bandwidth available in shared communication networks. Usually, the laboratory experiments are firstly analyzed and simulated on a Matlab/Simulink environment with its toolboxes, namely the Signal Processing, the Control System, the System Identification, the Image Processing and the Symbolic Math Toolboxes. Next, the user has the possibility of execute experiments on-site or remotely using the platform.

Some of those experiments for K-12 education, engineering and lifelong learning courses are briefly described as follows.

A. Experiment 1: Hydrostatic and hydrodynamic

The purpose of this experiment is to analyze and observe different aspects and laws of hydrostatic and hydrodynamic, at a secondary education course level, such as Physics. The two-tank system (model PCT9 from Armfield) is one of the systems considered in this experiment. This system (Figure 10) uses a motorized valve to control the flow rate of water pumped from a sump tank to a dual compartment tank at the top of the plant, where a sensor measures the liquid level. Several outputs with different flow rates can be chosen.

This process could be used to demonstrate the laws of hydrostatic and of hydrodynamic. In hydrostatic, the students can realize the notion of fluid, volumic mass or its fundamental law. In hydrodynamic, some experiments can be conducted to observe the movement of fluids, the laws of conservation of mass and energy, the Bernoulli’s equation or other laws of hydrodynamic.

B. Experiment 2: Networked Control Systems

This experiment aims to introduce a non-dedicated communication network between the controller and the system to build a networked environment, at an engineering course level. Using the Internet to remotely control a real system, it is possible to analyze the robustness of the overall control system to delays and faults in the communication network.

The three-tank system (model DTS 200 made by Amira) is a benchmark system that is used to implement a networked control system. This laboratory process (Figure 11) is a nonlinear multi input-multi output (MIMO) system with three level sensors and two actuators (pumps).
C. Experiment 3: Distributed Control Systems

In this experiment a Supervision and Distributed Control System (DCS) over a Wireless Sensor and Actuator Network (WSAN) is considered in the context of lifelong learning courses. Several aspects concerning supervision systems, wireless sensor and networks communication protocols are studied and a distributed control system is implemented and tested.

The diagram represented in Figure 12 shows a distributed control system based on the two-tank system referred in experiment 2 and a WSAN that reads some temperatures. The sensors and the actuator are connected through the nodes of the network to a gateway computer. A user can choose to implement the controller inside the network in the actuator node or outside considering a remote controller that interacts with the system through the gateway.

Given the distributed characteristic of this system, the user can analyze several aspects of the WSAN and implement local or remote controllers and supervisory control systems.

V. CONCLUSION

This paper described the current version of the remote and virtual control laboratory RVL@DEI-UC and its integration in an intelligent tutoring system, showing the adaptability and potentialities of available experiments, for example, in the context of secondary education, engineering and lifelong learning courses.

The main characteristics of the RVL have been explained and different types of remote and virtual real-time experiments succinctly described. Although our intelligent tutoring system is at an early stage of development, it was highlighted the relevance of the integration of the RVL in an intelligent tutoring environment, supporting the improvement and the adaptability of the learning process to different users’ profiles.

This paper aims also to contribute to highlight the relevance and potential of remote and virtual labs for the enhancement of the learning process in the classroom or in an e- and b-learning context.

ACKNOWLEDGMENT

The authors wish to acknowledge the support provided by the Centre for Informatics and Systems of the University of Coimbra (CISUC) for the development of this project.

REFERENCES