Available online at www.sciencedirect.com

ScienceDirect

Procedia Social and Behavioral Sciences 2 (2010) 5271–5275

ELSEVIER

Procedia
Social and Behavioral Sciences

WCES-2010

An internet-based real-time remote automatic control laboratory for control education

Mehmet Uğur[a], Kenan Savaş[a] *, Hasan Erdal[a]

[a] *Marmara University Technical Education Faculty , Istanbul, Turkey*

Received November 10, 2009; revised December 11, 2009; accepted January 21, 2010

Abstract

This paper describes a web based remote automatic control lab, RACL, system. This is a system that enables students do real time control experiments from a remote location (e.g. their homes) using internet and an ordinary web browser. By using this system one can select an experiment, launches that experiment, and examines real time experiment outputs by looking at charts. This system brings hope for reducing costs and improving control education quality.

Keywords: Automatic control; control education; remote laboratory; distance education.

1. Introduction

Studying in labs is crucial for technical education (Li et al., 2003). Experiments done inside labs gives real grasp of technical knowledge that taught at courses. Analysis, design, measuring, collecting data, interacting with real tools are main activities of lab studies (Ashby, 2008). Remote Lab is an environment that has a software basis and lets students do available experiments on available devices from a remote location (Deniz, Bulancak & Ozcan,2003). Remote Lab in some sense resembles to virtual labs. Virtual labs simulate behavior of an experiment by evaluating input and generating output programmatically (Hasnim & Abdullah, 2007). Using virtual labs resulted in increased learning effectiveness and evolves conventional education techniques to a new technique that puts the learner at the center of education activity (Aidan, 2009). However virtual labs fail to fully simulate real experiments. It is formally approved that results generated by Remote labs are valid by The Accreditation Board for Engineering and Technology Inc. (ABET) in 2002 (Joe, 2008). Interaction with real physical processes that is lacked by virtual labs makes remote lab systems more attractive (Casini, Prattichizzo, & Vicino, 2004).

Remote lab concept has a long historical back ground and can be traced to a remotely controlled lab project that is developed at the Argonne National Laboratory in 1954 (Joe, 2008; Esche, 2002). Examples of remote labs are developed by M. Casini (Casini, Prattichizzo, & Vicino, 2003) and by A. Khamis, M. Pérez Vernet, K. Schilling (Khamis, Vernet & Schilling, 2002), B. Doğan (Doğan, 2007; Doğan & Erdal, 2008). In future more elaborate

* Kenan Savaş Tel.: +90 5327304408
E-mail address: kenan.savas@marmara.edu.tr

doi:10.1016/j.sbspro.2010.03.859

remote control systems will be developed that are capable of delivering more precise outputs and present larger set of functionality (Humos et all., 2005).

Designed Remote Automatic Control Lab (RACL) to take benefits of Remote Lab systems and to avoid problems that are faced in typical labs . We were facing lack of space and instructor. Due to inadequate quantity of lab tools, groups of participants had to be large in number. However, this had leaded to reduced quality of education. Lab sessions were so short that students were getting hard times to complete experiments.

RACL is developed as a server client system. RACL has three main components. Client of the RACL system connects to the system using an ordinary internet browser and a Java Applet. Selects an experiment and specifies parameters then launches experiment. At the server side, server application gets user requests and delivers to a Control Program, developed using MATLAB (Savaş, 2007; MATLAB Web Server User's Guide), residing at the same server machine. The Control program controls laboratory devices according to parameters taken from server application. Once the experiment started output of the experiment devices are collected by a data acquisition card at real time. The data acquisition card sends output data to the Control Program. The Control Program evaluates the output data and sends to the Server Application. What next is Server Program sends output data to Java Applet running at remote user machine. This way, client can pick an experiment, set desired parameters and can observe output collected from devices by looking at graphs.

This paper is laid out as follows: Software Structure, Hardware Structure.

2. Software Structure

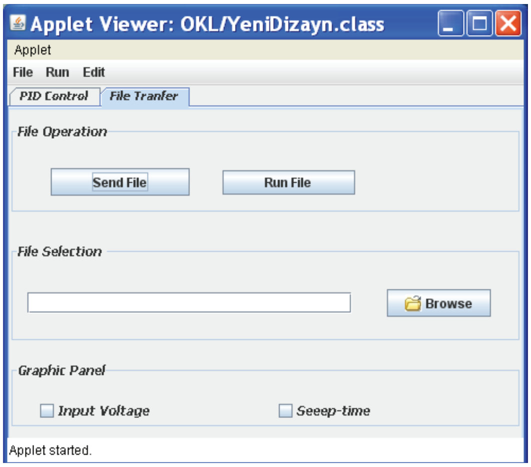
Software structure of RACL consists of two main parts: Client side and Server side.

2.1. Client side of the system

Using an ordinary web browser, user can access to automatic control lab. A Java Applet that runs on web browser connects to server machine by using a socket connection. This applet has two interfaces: Parameter setting interface, File transfer interface. Figure 3 represents the client side of RALC's flow chart.

2.1.1. File Transfer Interface

Using this interface user can select MATLAB file (M file) that can be used to specify a custom control model. Figure 1 shows File Transfer Interface. If received file is not suitable, it will not be accepted by our server.

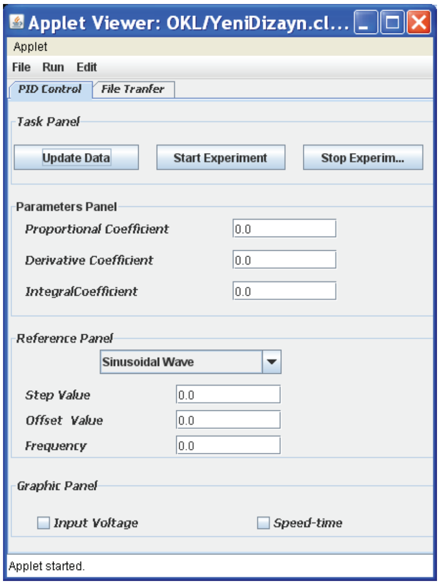


Applet started.

Figure 1. File Transfer Interface

2.1.2. Parameter setting interface

By default system uses PID control model for controlling experiment devices in a remote control lab. PID stands for Proportional Integral Derivative control model (Ogata, 2001). Control parameters and functions that are to be applied to entry of experiment devices are specified through this applet interface. Some examples applicable functions are Unit Step Function, Ramp Function, and Sinusoidal function (K. Ogata, 2001). Figure 2 shows Parameter setting interface.



Parameters Panel

Proportional Coefficient 0.0

Derivative Coefficient 0.0

IntegralCoefficient 0.0

Reference Panel

Sinusoidal Wave

Step Value 0.0

Offset Value 0.0

Frequency 0.0

Applet started.

Figure 2. Parameter setting interface.

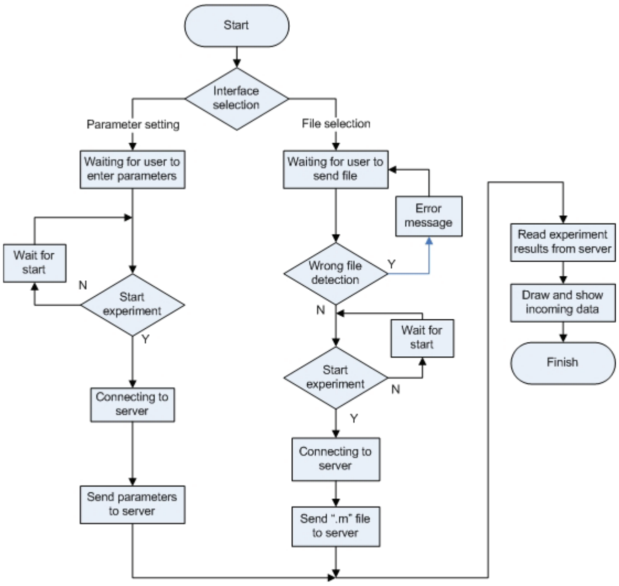


Figure 3. Flow chart of client side of RALC

Both interfaces can be used to start and stop an experiment, and a graph that shows output of currently running experiment can be switched on or off. This graph, showed at Figure 4, can be zoomed in and out to specify detail level of output data. This graph also shows numerical value and time value of a graph section that is under cursor. This yields a more user friendly interface.

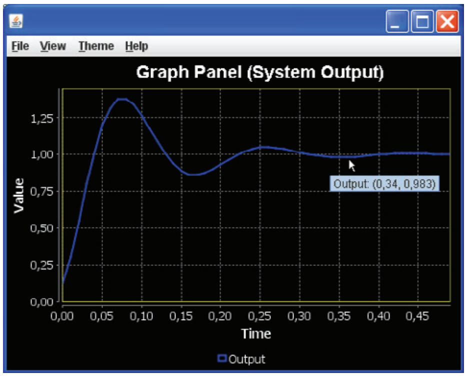


Figure 2. Graphic Chart Panel.

2.2. Server side of system

Server side consists of three components: Web Server, Java Server Application and control program that written in MATLAB.

2.2.1. Web Server

User at the very first step accesses to Web Server to request web page that Java Applet resides in.

2.2.2. Server Application

Once Java Applet is started in user computer, it sets up a socket connection with Server Application which is also developed using Java (Retrieved November 13, 2009, from http://www.sun.com/). This application is responsible for directing data and control flow between Client and Control Program. A user can at most stay in system for five minutes to give other clients chance to use system.

2.2.3. Control Application

This software is developed using MATLAB. Main role of the Control Application is to control experiment devices and collect output data that are delivered by Data Acquisition Card. Experiment devices are controlled according to parameters taken from Server Application. Data delivered from Data Acquisition Card is sent to Server Application. Hardware structure is shown in figure 5.

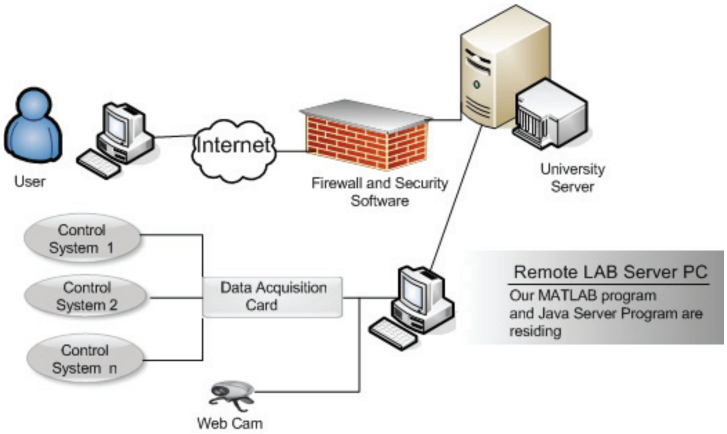


Figure 3. Hardware Structure of RACL .

3. Hardware Structure

Hardware of RACL has three main parts. These are Server Machine, Data acquisition card and Web cam.

3.1. Server machine

This computer is a host for software components of RACL system. Server machine is located inside the Control Lab inside our Faculty Building. This computer should be strong enough to serve a couple hundreds of clients. This pc must be secured against malwares and other threats.

3.2. Data acquisition card

This card is developed parallel to this project by other members of our faculty to support projects with the minimum cost. This acquisition card is connected between Server Machine and experiment systems. It is PIC based and connected to server machine via RS232 interface. Responsibility of the Data Acquisition Card is delivering control signals generated by control software to driver of experiment systems and collecting data produced by experiment systems in order to send to control software. Data produced by experiment systems can be at speed of an electrical Motor, liquid level control, pressure level control or temperature control system etc.

3.3. Web camera

Functionality of RACL system can be extended easily by adding a web cam support. Using this web cam one can see real experiment devices working.

4. Conclusion

Implementation of RACL was easy and cheap. We tried to use free software tools when possible. Use of Java was an element that reduced cost and development time. After realizing and utilizing RACL project we depleted problems of congestion, lack of instructors and lack of space. RACL diminished demand for buying additional Laboratory Devices by leading more effective use of available ones. RACL also took positive views of both students and educators. Students running experiments in a more relaxed and comfortable environment made a quiet well learning progress.

References

Aidan O'Dwyer (2009). School of Control Systems and Electrical Engineering, Dublin Institute of Technology, Kevin Street

Ashby Joe E. (2008). The Effectiveness of Collaborative Technologies In Remote Lab Delivery Systems. In 38th ASEE/IEEE Frontiers in Education Conference

Casini M., Prattichizzo D. ,& Vicino A.. The Automatic Control Telelab: A User-Friendly Interface for Distance Learning. Journal of IEEE Transactions On Education, Vol. 46, No. 2, May. 2003.

Casini M., Prattichizzo D., & Vicino A. (2004). The automatic control telelab: A Web-based technology for distance learning IEEE Control Systems Magazine

Deniz D. Z., Bulancak A., & Ozcan G. (2003). A novel approach to remote laboratories. In Proceedings of the 33rd ASEE/IEEE Frontiers in Education Conference, Session T3E. Boulder, CO: IEEE.

Doğan B., "System Control Through the Internet and a Remote Access Laboratory Implementation", Master thesis, Marmara University Institute for Graduate Studies in Pure and Applied Sciences, 2007

Doğan B., & Erdal H., "System Control Through the Internet and a Remote Access Laboratory Implementation", Lecture Notes in Computer Science, LNCS 4823, pp. 532-541, March, 2008.

Esche S. (2002). Remote experimentation - one building block in online engineering education. In Proceedings of the 2002 ASEE/SEFI/TUB Colloquium. Berlin, Germany: American Society for Engineering Education.

Hasnim H., & Abdullah M. Z. (2007). Remote Lab Generator (RLGen): A Software Tool Using Auto-Generating Technique to Develop a Remote Lab. In

Humos A. A., Alhalabi B., Hamzal M., Shufro E., & Awada W. (2005). Remote labs environments (RLE): A constructivist online experimentation in science, engineering, and information technology.In Proceedings of IECON 2005. 32nd Annual Conference of IEEIndustrial Electronics Society (pp. 2156-2161). Raleigh, NC: IEEE.

Java Programming Language, http://www.sun.com/java/ (13rd Nov, /2009)

Khamis A., & Vernet M. Pérez & Schilling K. (2002). A REMOTE EXPERIMENT ON MOTOR CONTROL OF MOBILE ROBOTS . In Proceedings of the 10th Mediterranean Conference on Control and Automation. - MED2002 Lisbon, Portugal, July 9-12, 2002.

Li, L., Lai, G., Wang, F., & Wu, F. (2003). Online autonomous guidance system for remote experiments in control. In Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 003: Vol. 3 (pp. 2444-2449). Washington, DC: IEEE.

Ogata K., Modern Control Engineering, Fourt Edition (2001), Prentice Hall

Savaş K., "Kontrol Sistemleri Için MATLAB'de GUI Uygulamaları Tasarımı.", Master Thesis, Technical Education Faculty, Marmara University, Istanbul, 2007 (http://posta.marmara.edu.tr/~kenan.savas/) (20th Nov, /2009)

The MathWorks Inc., MATLAB Web Server User's Guide., 2003, http://www.mathworks.com/access/helpdesk/help/toolbox/webserver/webserver.html. (13rd Nov, 2009)