IFAC

Virtual Laboratory for Distance Learning, Remote Process Control

Dumitru Popescu, Ion Matei,
Catalin Dimon

University "Politehnica" of Bucharest,
Department of Control Engineering,
Splaiul Independentei 313, 060042, sector 6, Bucuresti

(e-mail: catalin.dimon@gmail.com)

Abstract: This paper describes the philosophy behind the remote level control of a didactical platform. Issues related to the hardware and software architecture that allow remote access to the experimental platform using a simple Internet browser and some concepts about the control strategy used are presented. The virtual laboratory application is built primarily as an educational tool for distance learning, but these types of applications can be extended, with some considerations, to industrial process control.

Keywords: remote control, virtual laboratory, educational aids, control algorithm, digital control.

1. INTRODUCTION

Today the Internet is widely used for educational, commercial, industrial and personal purposes, as it represents a gate to a huge amount of information. From its birth, in the 1960, the Internet has evolved, from a structure offering only static information (from server to clients), to a structure that involves real-time information transfer, allowing connection to equipments or processes anywhere in the world. This type of architecture facilitates the distance learning technologies.

The possibility to interact with didactical platforms by Internet allows free access to laboratory activities, since students are able to perform their practical work without time or/and place limitations. Remote access reforms the traditional way of education by reducing the time spent in universities and by increasing the role of self education.

The remote control does not limit itself to education only. While individuals become more and more active because of the economy's dynamic, the remote access represents an interesting option for scientists who will have the possibility to control equipments spread over a large area. The advantages of remote control led to an important number of research activities. An example of a distributed laboratory built around object oriented protocol (TelRIP) is described in (Kondraske *et al*, 1993). Another example is a distributed monitoring and control system created by a university in Singapore (Tan *et al*., 2002).

This paper presents the hardware/software architecture of a level control system with the possibility of remote access using the Internet. The Internet is used as a communication medium because of its capability of covering large areas, thus assuring access to the platform from any place with an Internet connection and an Internet browser.

2. VIRTUAL LABORATORY

The experiment presented in this paper aims to familiarize students with the practical work regarding system identification and numerical process control. Students have to control the fluid level of a tank by testing different types of numerical control algorithms. The figure 1 presents the level control didactical platform used in the laboratory.
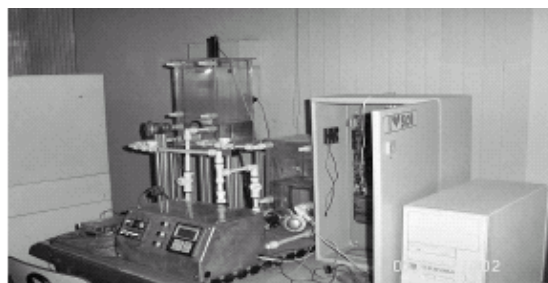

Fig.1. Level Control Didactical Platform

2.1. Level Process Control

In the virtual version proposed, the activity is structured in two parts. In the first part students practice the open loop identification of the studied process, the presence of the students in the laboratory being compulsory. The second part of the lab consists in digital control of the identified process by testing different types of algorithms. For this phase the presence of the students in the laboratory is not required because the activity can be undertaken from any place with an Internet connection (at home for example).

The practical activity of this virtual experiment is the level control of a fluid from a tank with consumer. Figure 2 presents the level control system schema, where r represents the set-point, u is the control and y the system's output.
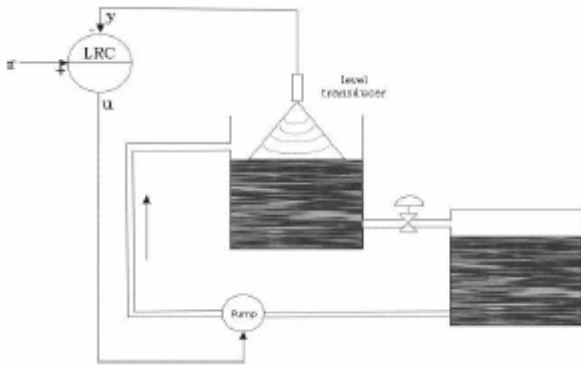
Fig.2. Schema of Level control system

The process input is the flow of water and is represented by a standard voltage signal (0..10V) that controls the power of a water pump. The pump controls directly the level of water in the tank depending of the consumer flow. The input signal is first generated in a digital form by the control software application, afterwards being converted in an analog signal by the acquisition board.

The process output is the level of fluid and is represented as a standard current signal (2..10mA) generated by an ultrasounds transducer placed on the top of the tank. The output signal is digitized by the acquisition board and then used by the control algorithm.

2.2. Hardware structure

The base of this experiment is a hardware structure composed of:

• Acquisition board with analog to digital and digital to analog converters which acquires and sends data to the process;

• PC which contain the acquisition board and hosts the control software application;

• Video camera for real-time broadcasting;
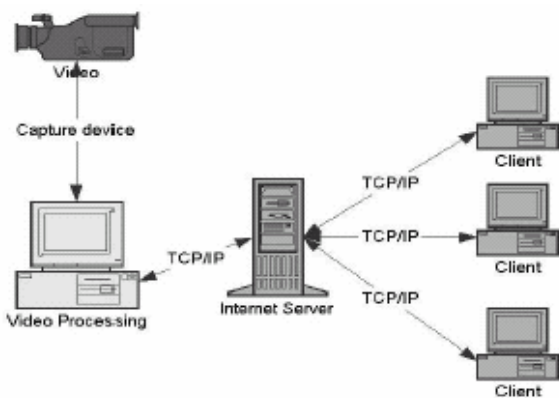
• Server for Internet connection.

Fig.3. Hardware structure for the virtual laboratory.

To make the control procedure more convincing a video camera is placed near the process. The camera broadcast live

images with the platform. In this manner the state of the platform, i.e. the level evolution, can be visualized remotely.

2.3. Software structure

The software structure is composed of two subsystems which allow access to the didactical platform locally as well as remotely, using the Internet, as it can be noticed in Fig. 4:
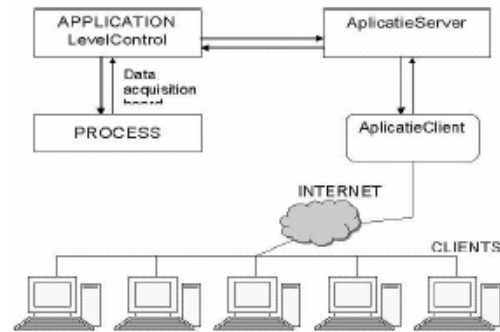

Fig.4. Software Control Structure.

The first subsystem represents a program written in Lab Windows/CVI. It runs on the computer connected to the process and allows direct local access to the platform. This application represents a gate for the data leaving or coming to or from the process respectively. The second subsystem is a client-server application that assures distance interaction with the process.

2.3.1 Local Access

The program for local access was created using Lab Windows/CVI and it is a stand-alone software control application that allows independent local access and control of the platform towards the remote access. For an easy use, the local program has a graphical user interface (GUI) presented in Fig. 5.
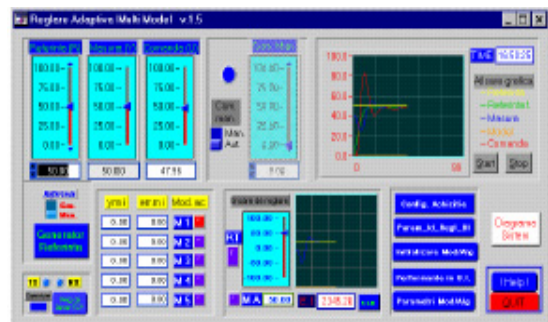

Fig.5. Local Access Interface

This program makes the connection with the process acquiring and sending data from and to the process respectively. The data is sent further to the client's applications via the Internet.

The local application assures a number of operations concerning the platform such as: set point changing,

automate/manual regime changing, manual command, control algorithms configuration.

2.3.2 Remote Access

The base of the distance communication is a *client-server application*. There are some keywords defining such an architecture which will be described as follows (Tannenbaum, 1999):

Server – entity which accepts requests and offers services to the clients. For this experiment the server is a program written in Java hosted by a server machine. The server connects itself to the local application and becomes a proxy for the clients connected to it. The server is a multithreading program that runs in background and periodically sends data about the state of the process to the clients. In theory it can accept an infinite number of client connections, but the number is limited by the software in order to maintain a certain level of performances and functionality.

Client – software entity which sends requests to the server and process the received data. For this experiment the client is a Java applet incorporated in a web page. Next to the applet there are presented live images with the process evolution. For security reasons imposed by the Java language the applet and the server must reside on the same machine. The client application has a graphical interface which presents in a natural manner information (received from server) about the state of the process and, in return, sends data to the process. For security reasons, the access to the web page requires an ID and a password.

Socket – a socket is an abstract object that represents a communication gate between processes in a network.

The client application offers the same facilities as the local application. The user can modify the set point, change the functioning regime (automate/manual) and generate manual commands as it is presented in Fig. 6:
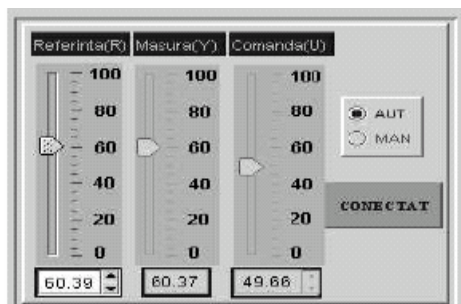

Fig.6. Interface for reference and command changing.

In the same time the user can see the time evolution graphic of reference, measure and command (see Fig. 7). Also, in order to increase the interaction with the platform, the user is able to design its own control algorithms and send them to the process for testing. For example, in Fig. 8 is presented the configuration window for a RST algorithm.
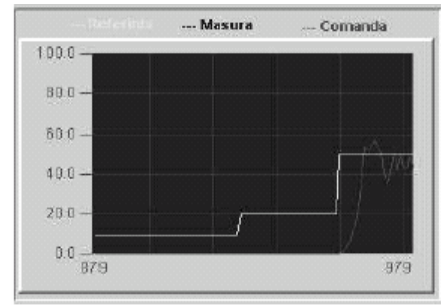

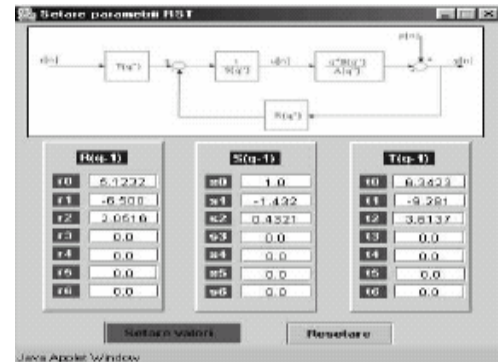Fig.7. Window presenting input, output and control evolution

Fig.8. GUI for RST parameters change

For security reasons, in case of a bad algorithm the local application will take over and replace the bad algorithm with the default one.

An important aspect is the study of delay due to network congestion. This delay can affect the update of the parameters by the client program or the receiving of the new data by the platform. However, because the data is sent at Te=5s rate, only a connection interruption could seriously affect the data transmission. The practice has demonstrated that a 56 Kb/s Internet connection is enough to assure adequate performances for the interaction with the didactical platform.

2.4. The access manager

An important issue related to the virtual laboratory is the access procedure to the platform, how is it done and what restrictions are imposed. In order to get permission to interact with the platform a student must first send a request (generally by e-mail) to the administrator. The student is then notified of the acceptance of his request, receiving a user ID and a password and a time window is set (taking in account the other activities involving the platform and the student requests) in which the student may perform the tasks involving the level control platform.

Only one student is allowed to remotely access the platform per one time window (an exception is made for the administrator that can intervene at any moment). The access manager is a database application which contains all the information regarding the access to the platform:

user ID, password, time window and day. Using this software tool the administrator performs all the necessary tasks for the schedule procedure. A log of all accesses to the platform is kept in order to have a clear image the remote activities.
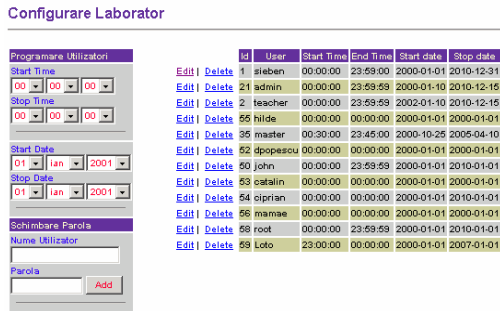

Fig.9. Access manager configuration window.

3. CONTROL STRATEGY

The process analyzed in this experiment is a nonlinear one. The nonlinearity is given by the relation between the level L and the flow F:

$$F = a\sqrt{2gL} \qquad (1)$$

In order to obtain a linear model, a representative point on the characteristic $F(L)$ was chosen for the identification process. In Fig. 9 the nonlinear dependence between the flow F and the level L is presented:
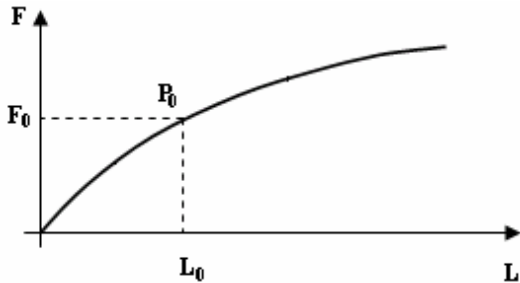

Fig.9. Plant input-output characteristic

Using a specialized software tool for system identification (PIM), the following discrete model was identified and validated, using as sample period Te = 5s:

$$\frac{B\left(q^{-1}\right)}{A\left(q^{-1}\right)} = \frac{0.184q^{-1} + 0.173q^{-2}}{1 - 0.325q^{-1} - 0.335q^{-2}} \qquad (2)$$

In order to design the control algorithm two steps must be followed: *the specification of the desired performances* and *the computation of an adequate control algorithm*.

For the control strategy, a RST numerical control algorithm (Fig.10) is proposed because it achieves the two main objectives of process control: *reference tracking and rejection of disturbances* (Landau, 1995; Gentil and Popescu, 1998).
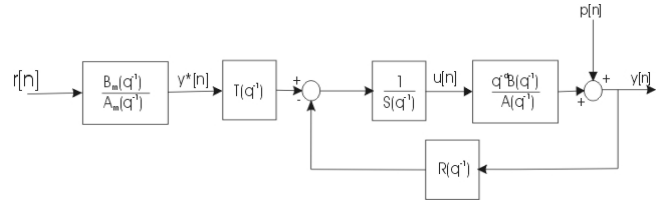

Fig.10. RST control algorithm with filtered reference.

The RST control algorithm has two freedom degrees and is described by the following expression:

$$S\left(q^{-1}\right)u[k] = T\left(q^{-1}\right)r[k] - R\left(q^{-1}\right)y[k], \ \forall k \in \mathbb{N} \quad (3)$$

It is easy to observe that this type of control allows to define the tracking and rejection performances independently, through polynomials $T(q^{-1})$ and $R(q^{-1})$ respectively.

The nominal performances are given by a second order standard dynamic system described by $\omega_0 = 0.05$, $\xi = 0.85$ (reference tracking performances) and $\omega_0 = 0.085$, $\xi = 0.75$ (disturbance rejection performances). With a sampling time Te=5s, we have obtained the following expression:

$$\frac{B_m\left(q^{-1}\right)}{A_m\left(q^{-1}\right)} = \frac{0.0071q^{-1} + 0.0066q^{-2}}{1 - 1.7669q^{-1} + 0.7807q^{-2}}, \qquad (4)$$
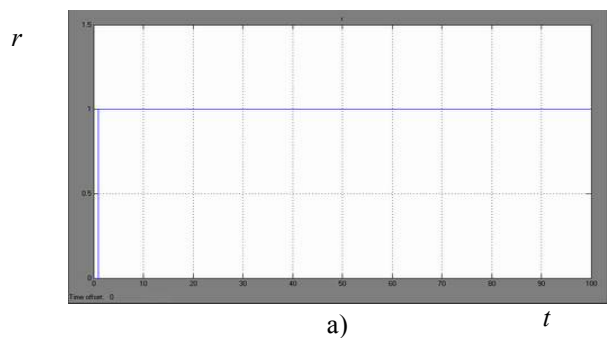
as reference model for the tracking performances and the characteristic polynomial:

$$P\left(q^{-1}\right) = 1 - 1.2807q^{-1} + 0.4493q^{-2}, \qquad (5)$$

which defines the rejection performances. The RST control algorithm satisfying the imposed performances is:

$$R\left(q^{-1}\right) = 5.1232 - 6.5q^{-1} + 2.0516q^{-2}$$
$$S\left(q^{-1}\right) = 1 - 11.432q^{-1} + 0.4321q^{-2} \qquad (6)$$
$$T\left(q^{-1}\right) = 6.3423 - 9.281q^{-1} + 3.6137q^{-2}$$

The system's response to step and the disturbance rejection are presented in the following figures:

r

a) t

$y*$

b) t

u

c) t

y

d) t

Fig.11. System response. a) reference - r; b) filtered reference - $y*$; c) command - u; d) output - y.

4. CONCLUSIONS

The structure presented here has been created as an educational tool for students in order to experiment remote access on a technological platform and perform comparative studies regarding the performances of different digital control algorithms.

By remote connection with the platform, using the Internet, a student is able to observe (real-time image broadcasting) the evolution of the process, to follow and change the state of the platform from any place, just accessing a web page (www.lab.uv.pub.ro). This manner of distance access transforms the traditional way of education into a more flexible and more personal learning experience.

The platform described in this article is used in a network of universities (Ecole Supérieure d'Ingénieurs d'Annecy, Polytech'Lille, etc.), created for promoting distance learning, as part of the European project: Socrates-PROG "Joint Development of Postgraduate Program in Electrical Engineering and Automatic Control".

REFERENCES

Kondraske, G. V., R. A. Voltz, D. H. Johnson, D Tesar, J. C. Trinkle and C.R. Price (1993). Network based infrastructure for distributed remote operations and robotics research. *IEEE Trans. Robot. Automat Name of book in italics or underlined* , vol. 9, pp. 702-704.

Landau I. D. (1995). *Identification et Commande des Systemes*, Ed. Hermes, Paris.

Popescu D. and S. Gentil (2005). *Commande Numerique et Intelligence Artificielle en Automatique*, Ed. Tehnica, Bucharest.

Tan K. K., T. H. Lee and C. Y. Sog (2002). Internet-Based Monitoring of Distributed Control Systems - An Undergraduate Experiment. *IEEE Trans. Education*, vol. 45, pp. 128-134.

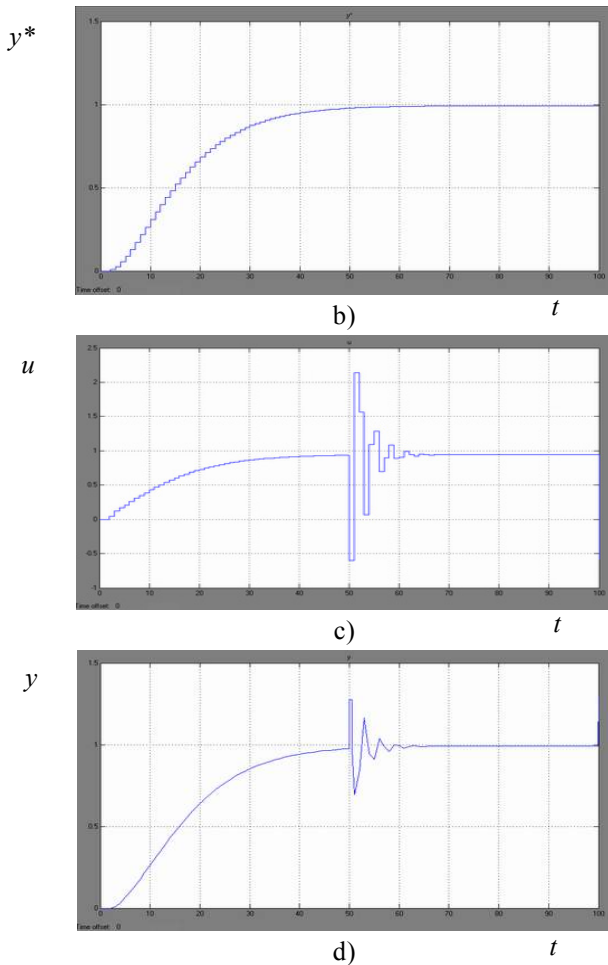Tannenbaum A. (1999). *Computer Networks*, Ed. Agora, Bucharest.