Robotica (2003) volume 21, pp. 303–312. © 2003 Cambridge University Press
DOI: 10.1017/S0263574702004812 Printed in the United Kingdom

Telecontrol system based on the Smith predictor using the TCP/IP protocol

Luis A. Baigorria P., José F. Postigo, Vicente A. Mut and Ricardo O. Carelli

Instituto de Automática (INAUT), Facultad de Ingeniería. Universidad Nacional de San Juan (UNSJ), Av. San Martin – Oeste – 1109, J5400ARL – San Juan (ARGENTINA).
e-mail: {lbaigorria, jpostigo, vmut, rcarelli}@inaut.unsj.edu.ar

(Received in Final Form: November 9, 2002)

SUMMARY
In this paper, development and implementation are presented of a client software package for remote process control. The proposed software is based on a client-server model under an Intranet architecture. The architecture is proposed for a telecontrol system of a real process, which includes the possibility of integrating I/O devices with data networks based on open protocols such as TCP/IP. This protocol allows the implementation of control systems using a low-cost alternative. Also, the Smith predictor is revised for remote control applications over an Ethernet network. Some experiences on a laboratory pasteurization plant are addressed to show both developed controllers and architecture performance.

KEYWORDS: Telecontrol; Communication protocols; Computer-aided control system design; Industry automation; Remote control; Industry automation; Time delayed systems; Control networks.

1. INTRODUCTION
One of the main characteristics of network-based control systems (NCS) is that they have one or more closed control loops via a communication channel. In NCS the research interest is focused on the control system stability and performance rather than on the communication network.

Even though networks play a subordinate role within the control system, they are still important, due to the fact that the feedback loop must be designed so as to use the communication channel as little as possible. The reason for this being that the network has multiplexed signals from other sensors and actuators. In addition, it features other communication possibilities associated to other tasks unrelated to control duties, as shown in Figure 1.

The control networks are typically LANs with a set of hierarchical networks above them.[1] The control loops are closed locally, by means of some remotely closed loops, such as in teleoperation systems[2] and monitoring systems with sensors distributed in WAN networks.[3]

Integration of computer networks into control systems in order to substitute the point-to-point wiring, has a number of advantages such as low cost, reduced power requirements, simple installation and maintenance and greater reliability. On account of this, the NCS are becoming a widely preferred option. In recent years, the discussion has been centered on the expansion of Ethernet uses into monitoring and control applications.[4] A considerable number of companies have began to use the Ethernet standard in their industrial devices.

However, the insertion of the network in the control loop makes the analysis and design of a control system based on networks a complex task,[5] due to the induced delay that occurs while the data is being exchanged. The delay can be constant or it can vary in time, degrading the performance of the designed control system. If the delay is not considered, it can make the response of the system unstable. In case the telecontrol system is implemented on the basis of the

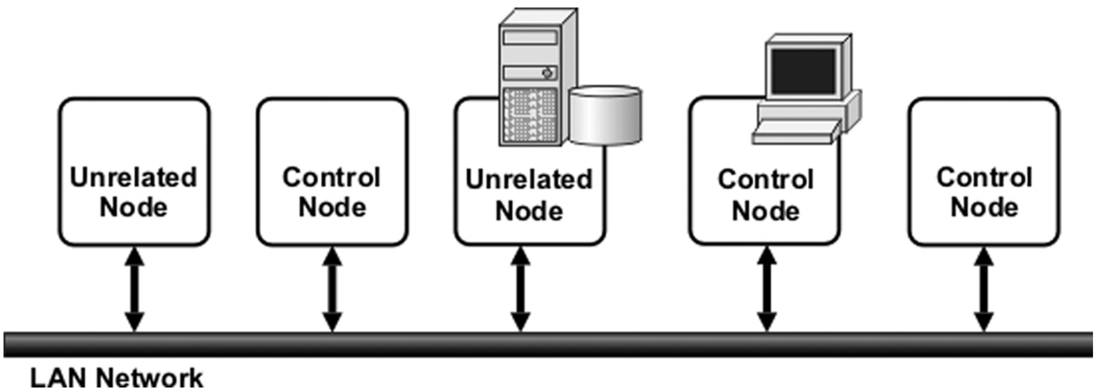

Fig. 1. Network-based Control System (NCS).

Ethernet network, the delay in the communication channel is arbitrary, and it varies with time, degrading the performance of the designed control system.

2. WORK OBJECTIVES

The main contribution of this paper is in the NCS field where an integration is proposed of the available tools and protocols with the Ethernet platform using TCP/IP. With this aim in mind, an architecture of a telecontrol system is developed with the TCP/IP-based communication protocol and the software architecture for a telecontrol application.

The application objective of this work is to telecontrol an available pasteurization plant by using a LAN with standard Ethernet, based on the TCP/IP protocol, without the need of implementing a purposeful data-communication system.

In addition, an adaptation of the Smith predictor is presented, adapted to be applied both to the remote control of industrial processes with inherent time delays, and to processes without inherent time delays. Some results from experiments are shown, which were carried out in the pasteurization plant.

The proposed developments are intended as a low-cost NCS implementation that enables to integrate sensors and actuators connected to a single data network for monitoring and controlling processes from various places through the Internet.

3. A LOW-COST OPTION

The communication channel is a LAN with standard Ethernet, based on the TCP/IP communication protocol. With the open architecture of Ethernet-based PC networks, it is possible to outline almost any conceivable system.

The high costs of proprietary networks and the need for a common standard network in the factory plants have led a great segment of the market to consider Ethernet for these applications. This tendency started with the use of Ethernet as a high-level control network to control different processes.

By applying these networks to new users of industrial equipment it is expected to have the same open architecture available, and the ability to combine equipment from different suppliers that share the environment.

TCP/IP is the most widely used protocol in Ethernet systems. Using a protocol that is common with the company information systems allows the data be easily and readily shared from the manufacturing areas into the R&D laboratories. It also enables to connect all levels of a corporate company, from the manufacturing plant to the administration and management floors, thus becoming a significant low-cost option.

Because Ethernet contains components that cost much less than the equivalent circuits of industrial communication, it has succeeded, in recent years, in industrial applications of process control.

3.1. Ethernet features

Prospective networks for control systems should generally meet two requirements: bounded delay time and guaranteed transmission. Ethernet uses a protocol known as Carrier Sense, Multiple Access, with Collision Detection (CSMA/CD) that impair the determinism.

However, Ethernet has some features that make it more advisable than other specialized industrial control networks. The wiring-type makes it easy to be installed and to expand. Low cost multi-port devices, called *hubs* are easily connected to the network backbone, the task of adding devices to the network thus becoming easier. The maintenance of an Ethernet is much cheaper and easier than any other network type. Ethernet is available in 10Mbps or 100 Mbps versions, providing a wide bandwidth, which is, translated into a faster response for real time applications.[6]

These features make the Ethernet a valid option to challenge the remaining control networks, which were developed to provide determinism in communication times, but pay a very high cost for this deterministic capability.

3.2. The TCP/IP protocol

Ethernet manages a great deal of communication protocols, but TCP/IP is the most broadly used one. Because of its high transmission rate and guaranteed distribution, the TCP/IP Ethernet system is ideal for monitoring applications and processing control from a remote site.

The TCP/IP protocol is widely supported as a communication protocol because it can be applied to almost any physical transmission media.[7] TCP/IP protocol allows different software protocols to co-exist with different hardware TCP/IP components, additionally providing an addressing system that enables one-to-one identification of each node over the remaining network nodes.

4. TELECONTROL SYSTEM ARCHITECTURE

The architecture proposed for the TCP-based telecontrol system is shown in Figure 2. The telecontrol system is composed of several client-server structures. In these structures, the clients are implemented by the software developed at the control PC, and the servers are at the nodes connected to the network that manage the I/O points.[8] There is a server at each network node. The nodes that manage the I/O points can be configured remotely, and they perform the updating of the values measured by the sensors. Besides, they are in charge of performing the control actions received from the clients at the output points.

In the control PC, there are clients available to carry out the communication with the server nodes. The PC software performs the control system by receiving the system response in real time, through the network and from the nodes. It also computes the control signals and transmits the control actions to the server. The server applies the control action at the corresponding point according to its configuration.

4.1. Advantages of the proposed architecture

Considering the proposed architecture, it is possible to implement servers with more economical devices and with lower power rates, but with lower computational power as well. The server tasks are limited to apply the control actions received from the clients, to keep updated the values of the variables measured by the sensors and to

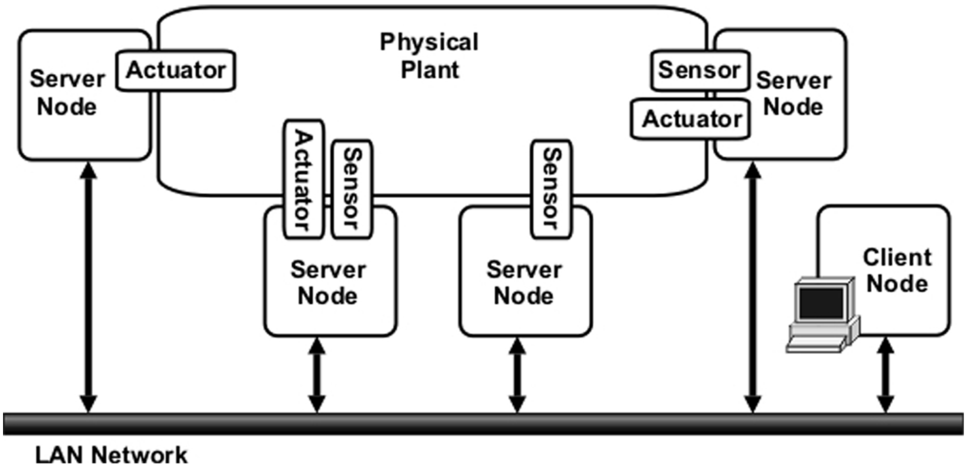

Fig. 2. Telecontrol System Architecture.

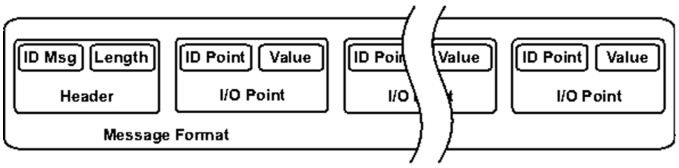

Fig. 3. Typical message format. This datagram has a variable length defined in the *length* field, indicating the number of points included in the datagram package.

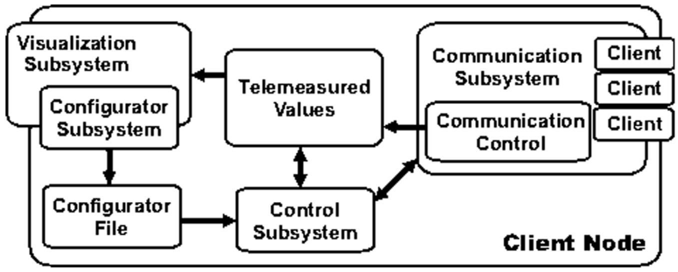

Fig. 4. Telecontrol application subsystems.

communicate the data to the client. This architecture enables the connection of many PCs operating coordinately via the network, thus permitting the computational power to increase for implementing complex control laws, independently of the server's limitations. Moreover, it provides greater flexibility in the design process of the controllers due to the freedom to implement the software by means of any controller type at the client.

4.2. Communication architecture

The TCP/IP communication architecture was developed for communication between the clients and servers, where the IP datagrams are exchanged among the nodes through the network layer. With the TCP/IP protocol, several clients with different IP numbers can be concurrently connected to the same server, and simultaneously demand data from it.

Datagrams format. Various message types were defined in order to be interpreted by clients and servers. A typical datagram transmitted between client and server is shown in Figure 3.

5. TELECONTROL SOFTWARE

With the tools available for object-oriented programming, such as the *Unified Modeling Language* (*UML*) and design programs such as Rational Rose®, it is possible to develop in a simple and orderly manner a powerful application composed of several subsystems to perform specific tasks.

UML allows the designer to develop several different types of visual diagrams that represent various aspects of the telecontrol system. *Use-Case diagrams* and *Class diagrams* shows interaction between classes, which represent the telecontrol system functionality. *Sequence diagrams* are used to show the flow of functionality through the communication process to perform the control task. *The State Transitions diagrams* are proper to represent the various states in which the object that perform the communication can exist. The *Deployment diagrams* show the physical layout of the network for the telecontrol system, and where the various components will reside.

Using *UML*, the designer can easily map the diagrams to *C*++ or Java code, then one can ensure that the requirements were actually met by the code, and the code can easily be traced back to the requirements.

The developed software is made up of several subsystems interacting with each other in order to perform the telecontrol tasks. It includes, basically, the subsystems for *configuration, control, communication and visualization*, as shown in Figure 4.

5.1. Configuration subsystem

When the user is carrying out the configuration of the telecontrol system, the system updates the information of the I/O points of the nodes through polling of each node, using a specific message. Each node returns a data package containing the information of the I/O point on which it acts. This information is stored in a file that will be used by other software subsystems.

The user begins the process by constructing or modifying the control structure that is stored in a file in order to be used in the control subsystem. The construction of the architecture begins by choosing the controller type to be used in the control loop. Then, the "input points" are chosen from a list, which will be used in the control algorithm. The same

https://doi.org/10.1017/S0263574702004812 Published online by Cambridge University Press

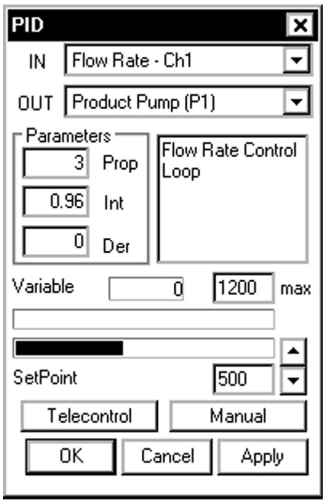

Fig. 5. PID Controller configuration window.

is done with the "output points", on which the control actions are to be performed. The loop's configuration ends by specifying the parameters of the controller according to the specific controller type, as shown in Figure 5. The configuration process ends when the user "adds" the controller to the file.

Once the package has been completed with the information from the control loops, it is used by the control subsystem.

5.2. Communication subsystem

The techniques used by telemetry systems to update the remote variables are applicable to the NCS as well.

The choice of data gathering technique used by the client can also limit the ability to use particular communication techniques.[9] The available techniques are:

- *Polling*: the *client* asks in sequence the data from each *server* node.
- *Polled Report by Exception Techniques*: the *client* asks only for the changes on the variables from *server* nodes.
- *Unsolicited Report by Exception Techniques*: the *server* node begins the transmission without this having been first solicited by the *client*. A background poll performed by the *client* verifies that data are updated and checks the "health" of remote devices.
- *Quiescent Techniques*: the *server* always begins the communication, the *client* (application) never asks information from any remote site. The most bandwidth efficient technique is the quiescent operation.

Among the above, the *Quiescent Technique* turned out to be the most appropriate one for a telecontrol implementation. In this, each server sends a message to the client in order to carry out the control action to be applied on a specific controller. As mentioned before, the method consists in sending the variables sensed in the server node, to the client at a sampling period rate specified by the controller. Once the datagram is received, the client sends out the variable with the communication subsystem, and "tells" the control module to perform the computation of the control action. Then the control action is transmitted to the node that has the variable on which it is intended to act upon. In this case, each of the *server* nodes has information about both the sampling period and the clients connected. Moreover, the server node sends the information periodically, according to the sampling period.

5.3. Control subsystem

The technique used for computing the control action is based on sending periodically the data from the nodes. Two developed subsystems are combined to perform this task: the control subsystem and the communication subsystem. The communication module is designed to receive data packages from the server nodes, to interpret the messages and to obtain the variables contained in the message. Once the variable has been identified, such an information is given to the control subsystem that executes the controller corresponding to the received variable from the communication subsystem. The control subsystem is in charge of executing the controller and points out the control action along with the identifier of the output point existing at some node server. The communication module determines the node such a variable belongs to, and sends the message through the network along with the control action to be applied to the corresponding actuator.

5.4. Server node architecture

The proposed *server node architecture* is simple, as shown in Figure 6. Its operation includes the possibility of accepting several clients that *subscribe* for the variables contained at the node, by using a specific message. The *server* node has a *timer* configurable from the client, in order to send the sensed variables that had been subscribed before. It also receives and performs the control actions at the corresponding output variables.

The *server* node electronic implementation includes a bus to connect the signal conditioning circuits for the sensors and actuators. The *server* node reads and writes through the bus the measured values from the signal conditioning circuits connected to it, keeping its complexity very simple. It is also expected to connect 4–20 mA loops drivers to the bus for communication with the sensors and actuators.

The *server* node also implements a second *timer* to detect communication failures, or the increase of the communication delay. When the *server* node receives control actions, it resets the *timer* value. If the *timer* finishes, the *server* node sets the output variables with secure operation values, and waits for the client to send new control actions or to reset the communication. The secure operation values and the

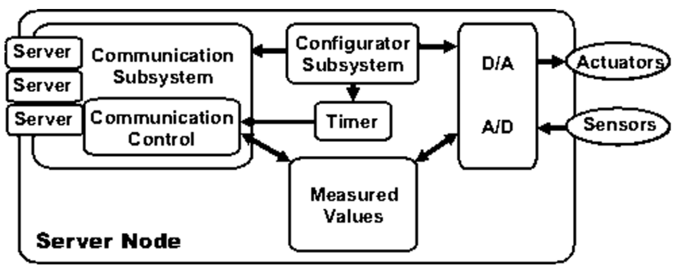

Fig. 6. Server node architecture.

https://doi.org/10.1017/S0263574702004812 Published online by Cambridge University Press

period of time are configured for every output variable from the *client* node, by using a specific message.

6. THE SMITH PREDICTOR

The presence of the delay in the system makes its control a more complex task, because the disturbances that enter in the process will not be detected after a long time. In addition, the control action taken, which is based on the last measurement will be inadequate because it will try to regulate a situation that happened a time before. This problem becomes worse by the fact that the control actions will also take a certain time in producing an effect on the process. As a result of all these mentioned factors, the dead time is important, and it is a source of uncertainty for the answer of the control loop. However, in the case of processes like those of pasteurization, the dead time is fundamental to achieve the production objectives. The Smith predictor is considered next.

Given a process that has an inherent delay time of d seconds, as that shown in Figure 7(a), Smith proposed to replace the controller C(s) by a new controller C*(s), such that in the closed loop, the system behaves as a process without delay, as shown in the Figure 7(b).

By equating both systems of Figure 7, equation (1) is obtained.

$$\frac{Y(s)}{R(s)} = \frac{C^*(s)\,G(s)\,e^{-ds}}{1 + C^*(s)\,G(s)\,e^{-ds}} = \frac{C(s)\,G(s)}{1 + C(s)\,G(s)}e^{-ds} \quad (1)$$

The Smith predictor equation is obtained as:

$$C^*(s) = \frac{C(s)}{1 + C(s)\cdot G(s)\cdot(1 - e^{-ds})} \quad (2)$$

Equation (2) is replaced into Figure 7(a) becoming the scheme of the Figure 8 that represents the basic configuration for the Smith predictor. The typical problems of implementing the Smith predictors include the estimation of the exact time delay d and the model of the process without delay G(s). Smith predictor can be applied, for example, to control the product temperature at the exit of the isolation tube in a pasteurization plant, because this tube introduces a delay into the system (see Figure 13). Then, to control the

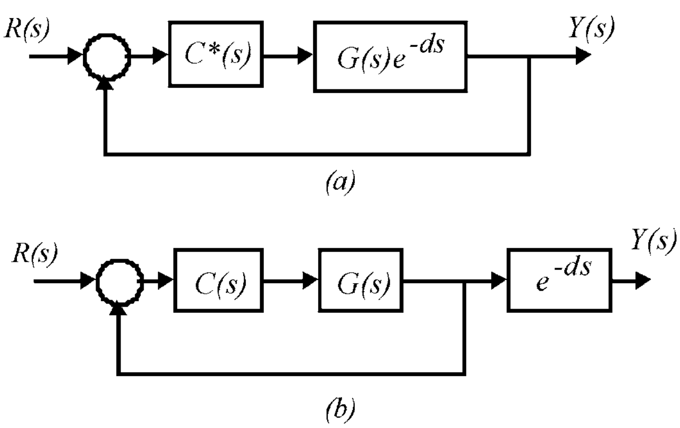

Fig. 7. (a) Basic system with the Smith predictor, (b) Equivalent system.

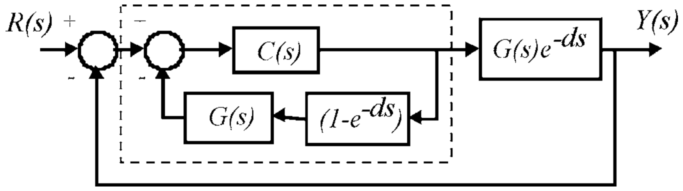

Fig. 8. Smith predictor architecture.

system remotely, the scheme of Figure 8 should be reviewed.

6.1. *Smith Predictor Applied to Telecontrol*

The Smith predictor proposed in Woods et al.[10] for remote control of systems introduces a variant in the Smith predictor that is not applicable to the telecontrol architecture outlined in this work: the communication channel is located between the controller and the plant. The Smith predictor is applicable to the architecture shown in Figure 9, and it is based on the theory proposed by Smith for the control of systems with inherent dead times.

A first consideration is that the communication channel produces the delay. The telecontrol system outlined is presented in the Figure 9(a). The local and remote terminals are physically separate, in such a way that the control undergoes a time delay m when transmitting the signals from the controller to the actuator, and a feedback delay n from the sensor to the controller. The use of the Smith predictor $C^*(s)$ in the local terminal, is aimed at eliminating the time delays within the control loop as indicated in Figure 9(b).

By equating both systems of Figure 9,

$$\frac{Y(s)}{R(s)} = \frac{C^*(s)\,G(s)\cdot e^{-ms}}{1 + C^*(s)\,G(s)\cdot e^{-ms}\,e^{-ns}} = \frac{C(s)\,G(s)}{1 + C(s)\,G(s)}e^{-ms} \quad (3)$$

Then, the Smith's predictor equation for telecontrol is,

$$C^*(s) = \frac{C(s)}{1 + C(s)\cdot G(s)\cdot(1 - e^{-(m+n)s})} \quad (4)$$

By replacing equation (4) in the scheme of Figure 9(a) the block diagram of Figure 10 is obtained. In Figure 10, the Smith predictor modifies the error signal through the

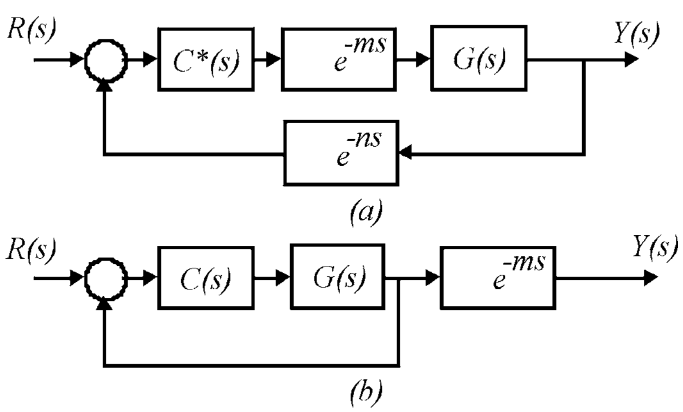

Fig. 9. Smith predictor for telecontrol of processes with delays in the communication channel: (a) basic remote control system, (b) equivalent system.

https://doi.org/10.1017/S0263574702004812 Published online by Cambridge University Press

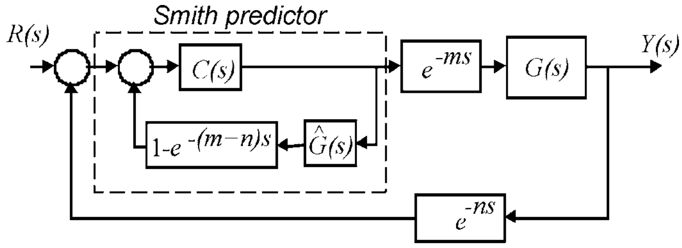

Fig. 10. Remote control of a process using a controller based on the Smith predictor.

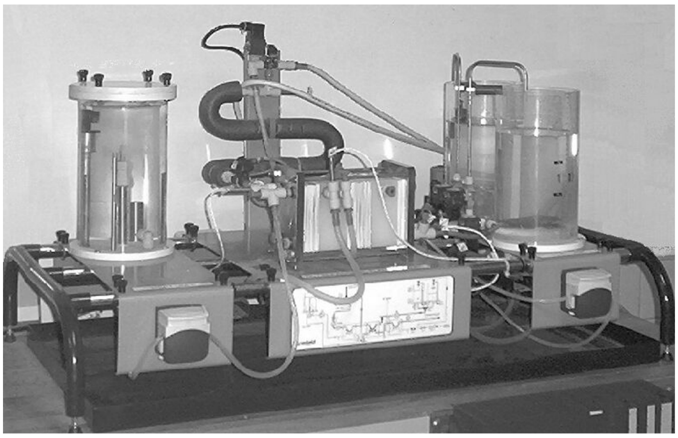

Fig. 12. Pasteurization process plant.

estimated output of the plant and through the delayed output of the plant, retarded an equivalent time to the delay that took place in the communication. The Smith predictor applied to the telecontrol of processes requires knowledge of the exact model of the plant and the communication delay of the loop $(m + n)$.

If the process has an inherent dead time, the Smith predictor of the equation (4) should be reconsidered. These modifications are studied in the following section.

6.2. Smith predictor applied to telecontrol of systems with inherent dead time

An isolation tube located at the output of a heat exchanger introduces a delay in the measurement of the product's temperature. This delay is necessary to complete the objectives of the pasteurization process. The remote control scheme of this loop is shown in Figure 11.

By repeating the same procedure of previous sections, we obtain,

$$\frac{Y(s)}{R(s)} = \frac{C^*(s)\,G(s)\,e^{-(m+d)s}}{1 + C^*(s)\,G(s)\,e^{-(m+n+d)s}} = \frac{C(s)\,G(s)}{1 + C(s)\,G(s)}\,e^{-(m+d)s} \quad (5)$$

Starting from equation (5) an expression for the Smith predictor can be obtained as:

$$C^*(s) = \frac{C(s)}{1 + C(s)\,G(s)\,(1 - e^{-(m+n+d)})} \quad (6)$$

In this case the delay used in the Smith predictor includes the inherent delay of the system (d) and the communication delay $(m+n)$. Therefore, the implementation of the controller needs to know the model of the process $G(s)$, the inherent delay of the plant, and the communication delay.

7. EXPERIMENTATION

A laboratory scale plant was used in the experiments. It consists on a pasteurization process incorporating a complete range of methods and strategies of process control: from simple feedback loops up to complex cascaded loops, with multiple actuators and sensors, as shown in Figure 12.

The plant operates with a high temperature-short time (HTST) industrial process for pasteurization that subject the product to a fixed high temperature for a short time, to suppress bacteriological agents. This is accomplished through an isolation tube that delays the product flow rate. This operation exposes particular control problems such as *dead time*. The plant uses a three-stage heat exchanger (recycle – heating – cooling) and a diverting valve to reject the improperly treated product. The plant introduces real control problems, thus allowing to test various control strategies. The plant scheme is shown in the Figure 13.

Some experimental results were obtained from controlling the described pasteurization plant by using local and remote controllers connected within an Intranet at INAUT laboratory. The first results allowed conclusions to be obtained about the effects of the communication delay on the control loop and, thus, to test the proposed application in order to perform future developments.

Here two control loops are considered that use the Smith predictor obtained in Sections 6.1 and 6.2. The structure of the telecontrol system and of the process components is

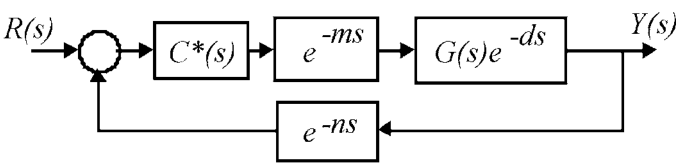

Fig. 11. Telecontrol of systems with inherent dead times.

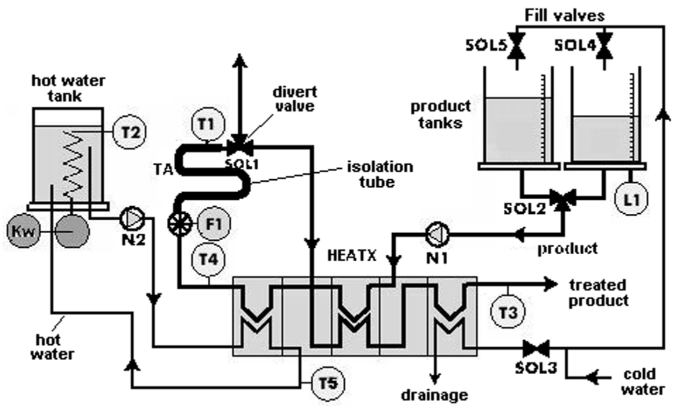

Fig. 13. General outline of the laboratory pasteurization plant.

https://doi.org/10.1017/S0263574702004812 Published online by Cambridge University Press

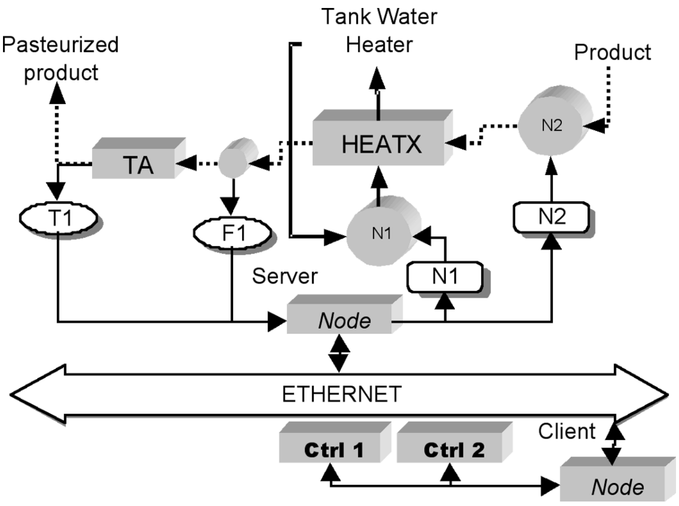

Fig. 14. Diagram of the telecontrol for the pasteurization process.

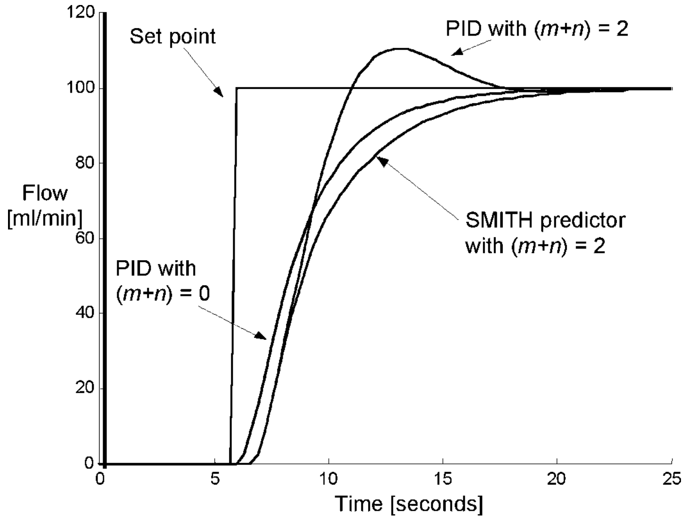

Fig. 15. Local control simulation with a PID, remote control with a PID and remote control with Smith's predictor with a delay of 2 sample periods.

shown in Figure 14. The communication channel is an Ethernet network on which the telecontrol experiments are carried out, thus fulfilling the objectives of controlling the pasteurization process. At the *client* node there is the application developed in the INAUT. This application determines at all times the communication delay.

The first control loop is applied to control the product flow (F1) acting upon the pump (N2). This control loop ensures a constant production rate whose maximum value is limited by the necessary time to keep the product inside the isolation tube (TA). This flow is 200 ml/min, for the plant.

The second control loop is applied to keeping the pasteurization temperature (T1) by acting upon the pump (N1). This loop ensures that the pasteurization temperature of 71°C for the product is fulfilled, while circulating within the isolation tube (TA). This loop contains an inherent delay in the temperature (T1) due to the presence of the isolation tube (TA).

The experiences performed are described next.

7.1. Experiment 1: Smith predictor applied to telecontrol

This experiment seeks to obtain conclusions regarding the use of Smith predictor as applied to the telecontrol of processes without inherent dead times. To control the product flow (F1), the plant has a turbine sensor installed in the product pipe as shown in Figure 14. This sensor measures the product rate in *ml/min* as it enters to the isolation tube.

The discrete transfer function that relates the voltage applied to pump (N2) with the measured flow (F1), with a sampling period of 0,3 *sec*, is:

$$\frac{C(z)}{E(z)} = \frac{1.1685z^{-1} + 1.22z^{-2}}{1 - 1.233z^{-1} + 0.3799z^{-2}} \quad (7)$$

The digital PID controller's equation is implemented by means of the recursive algorithm of Equation (8).

$$u(k) = u(k-1) + A \cdot e(k) + B \cdot e(k-1) + C \cdot e(k-1) \quad (8)$$

The values of constants A, B and C in Equation (8) are:

$$A = K(1 + T_d/T_o) \quad (9)$$

$$B = K \cdot (1 + 2T_d/T_o - K_i \cdot T_o) \quad (10)$$

$$C = K \cdot T_d/T_o \quad (11)$$

where constants K, K_i and T_d are respectively, the proportional, integral and derivative constants, and T_o is the sampling period (0.3 sec) used.

Figure 15 shows the results from simulations of the pasteurization plant telecontrol carried out with MATLAB. The simulation considered the remote control of the plant using the traditional PID controller given by Equation (8), without compensating the delay of $(m+n) = 2$ sample periods with constant $Kp = 3$, $Ki = 0.96$, $Kd = 0$ and $To = 0.3$ seconds. In Figure 15 it is clearly noted that the answer of the system is degraded when the delay increases; the settling time and the response overshoot also increasing. For great delays the system becomes unstable.

By using the Smith predictor of Equation (4), it is noted in Figure 15 that the answer of the system approaches the case of the system without delay, displaced on the time axis by a time equivalent to the communication delay.

Experimental results. Figure 16 shows the response when using a PID controller to control the flow (F1) at the same place where the pasteurization plant is located (local control).

The experimental results are shown next, when the control of the process is executed from the remote site by using the same PID controller of Figure 16. Figure 17 shows the answer obtained and the delay measured at the communication channel during the experience. An increase of overshoot is noted regarding the answer of Figure 16, due to the presence of the delay (of approximately a sample period – 300 *mseg*), which is not compensated.

Finally, the remote control experiences were carried out by controlling the flow (F1) through the Smith predictor.

https://doi.org/10.1017/S0263574702004812 Published online by Cambridge University Press

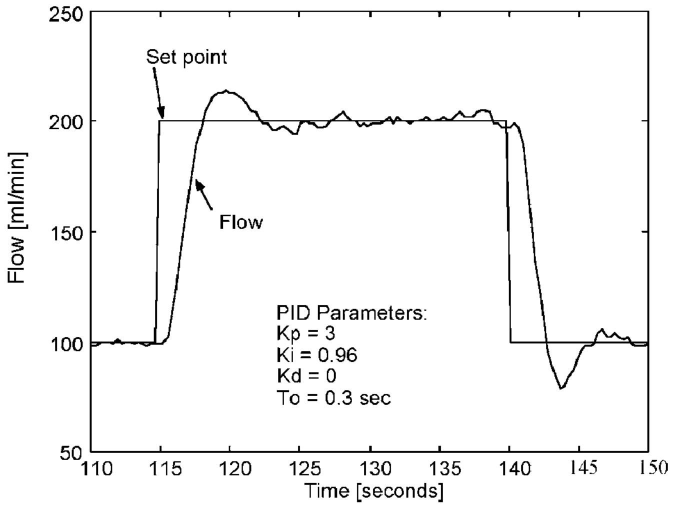

Fig. 16. Local control of flow (F1) with a PID controller.

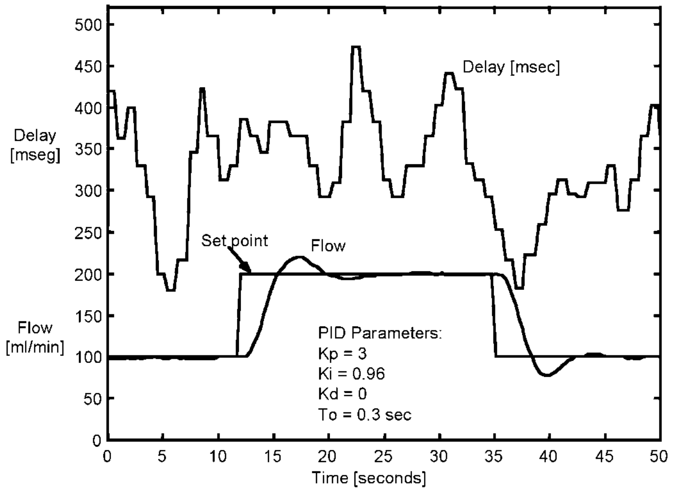

Fig. 18. Remote control of flow (FL) with the Smith's predictor.

The Smith predictor is built based on the PID controller of previous experiences. Figure 18 shows the system response and the communication delay measured during the experience. The conclusions obtained when observing the controller response compensating the delay is a response with smaller overshoot and less oscillations, and an improvement on the performance of the system of Figure 17, as well.

7.2. *Experiment 2: Smith predictor applied to telecontrol of systems with inherent dead time*

In this experiment, the Smith predictor is applied to the telecontrol of a process with inherent dead time aiming at reaching some conclusions about it. The requirement for this control loop is to ensure a temperature (T1) higher than 71°C, measured in the output of the isolation tube (TA). The Smith predictor used for this case is expressed in Equation (6).

The discrete transfer function that relates the temperature (T1) to the voltage applied to pump (N2) which uses a 1 *sec* sampling period, is:

$$\frac{C(z)}{E(z)} = \frac{1.685z^{-1} + 1.22z^{-2}}{1 - 1.233z^{-1} + 0.3799z^{-2}} \tag{12}$$

Experimental results. Figure 19 shows the results obtained when applying the Smith predictor of Equation (2). That controller does not include the communication delay. In the experiment, the product flow rate was 200 *ml/min*, which produces a 20 seconds delay. It is noted that the answer of the system controlled with the Smith predictor, remains in time.

The figures depict the prediction carried out with the model approaches to the real answer finally obtained.

In Figure 20 the controller's answer is shown when the remote control of the process is carried out using the Smith predictor of Equation (6). The answer of the system is noted to be acceptable, remaining controlled about the 72°C.

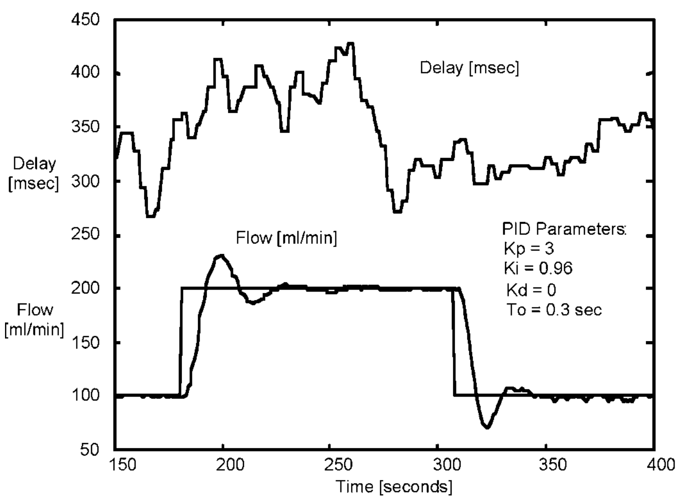

Fig. 17. Telecontrol of flow (F1) with a PID controller, without delay compensation.

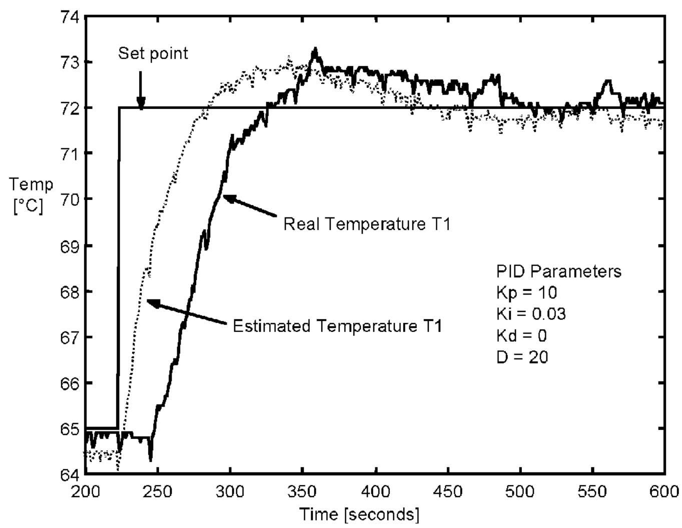

Fig. 19. Local control of temperature with Smith's predictor.

https://doi.org/10.1017/S0263574702004812 Published online by Cambridge University Press

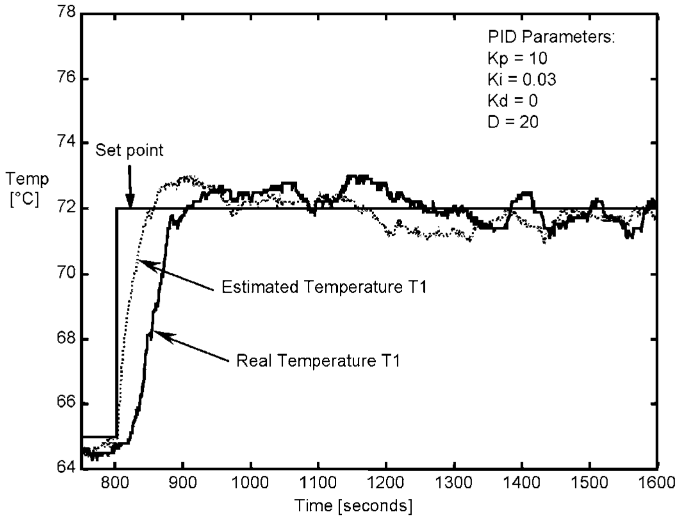

Fig. 20. Remote control with Smith's predictor applied to the telecontrol of systems with inherent delays.

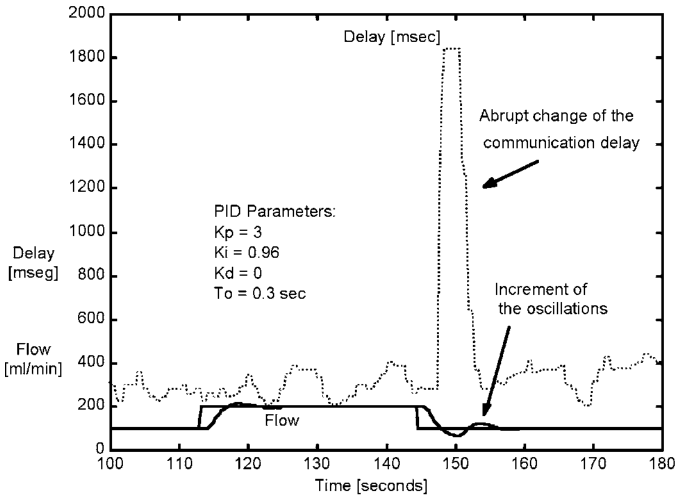

Fig. 21. Variation of the communication delay when a change in the reference of the loop is controlled.

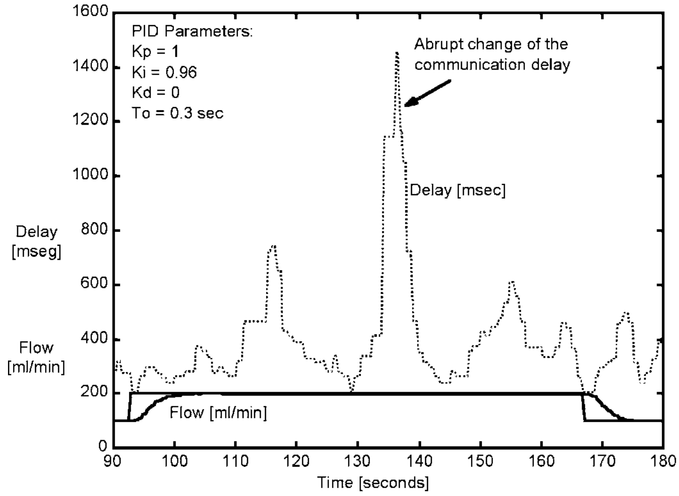

Fig. 22. Variation of the communication delay in the stationary state of the control system.

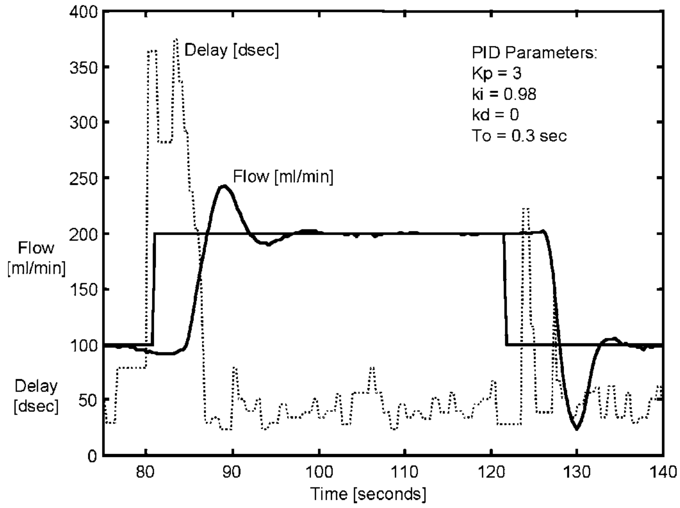

Fig. 23. Variation of the communication delay before the change of the reference of the system.

7.3. Experiment 3: Telecontrol of process with variable communication delay using the Smith predictor

Finally, an experiment is carried out on the effect of the delay variations in the control loop by introducing interruptions in the communication.

Figure 21 shows an increase of the communication delay of up to 6 sample periods (1800 *msec*) at the instant the Smith predictor is controlling the reference change of the flow control loop (F1). This delay produces an increase in oscillation, which is finally controlled.

Figure 22 shows the result when a change is introduced in the delay of 5 sample periods (1500 *msec*), when the system is under steady state. The effect that takes place in the controller control action is negligible.

Finally Figure 23 shows the response of the system to the variation in the communication delay, in the instant before a change in the reference of the control loop takes place. The communication delay varies sharply from one to 11 sample periods. It may be observed that the system does not respond during the time of the variation of the communication delay.

8. CONCLUSIONS

The paper has presented the development of a telecontrol system of a lab pasteurization plant based on an Ethernet network as a communication channel. The feasibility of using the TCP/IP protocol for remote process control has been tested.

Regarding that a real process was used instead of a PC-simulated process, it was shown that the client-server model is an appropriate approach for the telecontrol of a process as a low-cost alternative.

The methodology presents the advantage of an easy and ready implementation as compared to other control schemes such as parameter tuning of the on-line controller at the client, while the server is transmitting the system's response.[11]

Also, the Smith predictor proposed for the telecontrol of processes and the Smith predictor implemented for the telecontrol of processes with inherent dead time, are controllers that have shown to have good performance for use in remote control of processes. The main disadvantage the Smith predictor shows is that both the exact process

https://doi.org/10.1017/S0263574702004812 Published online by Cambridge University Press

transfer function and the delay between the local and the remote sites must be known.

On the other hand, it has been demonstrated experimentally that the use of the Ethernet network for remote control introduces a variable delay that degrades the behavior of the system. When the system is in the steady state or close to it, the variations that take place do not have great influence on the response of the system. However, the variations that arise when the system is controlling a reference change in the control loop are critical. In such a case, the solution may be to reduce the gain of the system by attenuating the effect of the variable delay in the behavior of the control loop, but sacrificing the settling time of the answer.

Future works plans to incorporate some controllers that take into account the uncertainties in the systems transfer function, the uncertainties in the communication delay, and the variations in the communication delay. The robust controller outlined in Landau et al.[12] takes into account the variations in the system delay. In addition, Olbrot et al.[13] present a robust controller that take into account the uncertainties in the delay of the system.

ACKNOWLEDGEMENTS

This work was partially financed by the National Council of Scientific and Technical Research (CONICET), Argentina and by a grant of the ANPCyT, Argentina.

References

1. D. Sonderman, “Getting Control of the Process”, *Sensors*, 12–18 (Oct., 1998).
2. A.C. Castro, J.F. Postigo, J. Manzano and B. Kuchen, “Sistema de Teleoperación de Robots con realimentación de fuerzas basado en redes TCP/IP”, *Proc. of the XVII Congreso Argentino de Control Automático (AADECA ’00)*, Bs. As, Argentina, (September, 2000) pp. 201–206.
3. A.C. Castro, E.M. Zavalla and M.L. Martín, “Modelo Cliente-Servidor para la administración visual en edificios”. *XVI Congreso Argentino de Control Automático (AADECA ’98)*, Bs. As., Argentina. (August, 1998), pp. 543–548.
4. T. Laine, “Impact of Internet technologies on industrial fieldbuses”, *Proc. of the 4th IFAC International Symposium SICICA ’2000* Bs. As., Argentina (2000), **Vol. 1**, pp. 57–62.
5. W. Zhang, M. Branichy and S. Phillips, “Stability of Networked Control Systems”, *IEEE Control Systems Magazine* **21**, No. 1, 84–99 (2001).
6. R. Winkler, “Clearing Up the confusion on Ethernet”, *Sensors*, 28–35 (January, 1999).
7. T.A. Lutz, “Using TCP/IP as an instrument interface”, *Sensors*, 43–46 (July, 1998).
8. D.E. Comer and L.D. Stevens, *Internetworking With TCP/IP: Client-Server Programming And Applications.* (AT&T TLI Version. Department of Computer Sciences Purdue University. Ed. Prentice Hall International Inc., 1994).
9. P. Aubin, “Telemetry Systems, Techniques and Standards”, *Instrumentation and Control Exhibition Symposium,* (June 1997). Available at http://www.iica.org.au/letters/telem.html.
10. H.C. Wood and M.R. Olson, “Time-Delayed remote control systems using Smith controllers”, *Proceedings of the IASTED Internacional Conference Intelligent Systems and Control*, Halifax, Nova Scotia, Canada (1998), pp. 36–39.
11. D. Gillet and E. Gorrochategui, “Remote Experiment with LabView”, *National Instrumentation Newsletter* **10**, No. 3, 6–7 (1998).
12. I. Landau, “Robust Digital Control of Systems with time delay (The Smith predictor revisited)”, *Proceedings of the 33rd Conference on Decision and Control (CDC ’94)*, Lake Buenavista, FL (1994) pp. 865–868.
13. A. Olbrot and M. Berri, “Perfectly Robust Deadbeat Controller for Systems with Unknown Delays”, *Proceedings of the 35th Conference on Decision and Control (CDC ’96)*, Kobe, Japan (1996) pp. 3190–3191.