

Simplifying the Practical Approach of the Process Control Teaching [★]

Ruben Morales-Menendez ^{*}
Tomas López, Ricardo A. Ramírez, Federico Guedea Elizalde ^{**}

^{*} *Tecnológico de Monterrey, Monterrey NL, México, rmm@itesm.mx*
^{**} {A00539053, ricardo.ramirez, fguedea}@itesm.mx

Abstract: Process control teaching at *Tecnológico de Monterrey* combines advanced didactic techniques, software systems and industrial equipment as instructional tools to achieve educational objectives. A software tool was designed to assist students in achieving deeper understanding on process control with practical experience. This software tool has the capability to exploit several real-time process control strategies implemented in *Matlab/Simulink* or compiled on *ANSI C* scripts. Early results showed that students can learn faster.

Keywords: Control Education, Linear Control Systems, Control Design, Computers for Control.

1. INTRODUCTION

Automation has become more important in the process industry in recent years. It is projected that the world market for process automation grows 5.1% annually from 2004 to reach \$ 94.2 billion in 2010 Schroeder [2003]. Also, the share of external engineering demand will keep increasing worldwide. It is expected to rise from 13.9% in 2000 to 15.5% in 2010. The trend towards rationalization and plant optimization, accompanied by increased integration of automation systems with the information systems of the production site and the enterprise level, will further add to the engineering share out of the total project costs, Jämsä-Jounela [2007].

More capable students are demanded today in the global competitive environment. Hiring engineers search for students who possess critical thinking, problem-solving, assertive communication, and other skills that complement technical capabilities. These skills can be enhanced by practical teaching systems.

Usually, control theory courses at universities are based solely on computer simulation. This educational model does not consider real situations, so students work on ideal, theoretical situations rather than practical systems. Many professors have tried to incorporate real process experience, but limitations (such as time constraints) difficult the implementing details. In many cases, additional background is needed in order to exploit experimental approaches.

In order to fulfill these needs, an educational proposal is hereby presented. This proposal considers a software tool that relates an industrial *PID* controller with a personal computer. This tool, follows as a "plug and play" philosophy, which avoids the training and time that most experimental laboratories demand. This interface allows

students to interact with real process systems, and to concentrate their attention on only the control engineering program.

This paper is structured as follows: In Section 2, a state of the art review of similar projects is presented. In Section 3, the academic proposal is described. In Section 4, results of undergraduate senior students are presented. In Section 5, a comparison with commercial solutions is included. Finally, conclusions and future work end this paper in section 6.

2. STATE OF THE ART

Although many significant and similar projects exist, the ones described below are related to the continuous control systems domain. Based on their main contributions, these projects were classified in three categories: simulation, laboratories and remote laboratories.

2.1 Simulation/Software

Several researchers consider that students can learn operation and design control strategies through dynamic models Cox et al. [2006]. Some interesting examples are shown next.

Cooper and Dougherty [1999] introduced Control Station (*CStation*), a computer software in which students can build and simulate dynamic systems (i.e. gravity drained tank, heat exchanger, jacketed reactor, distillation column, etc.) as well as different control strategies. It also includes design tools.

Irawan et al. [2001] presented a process system website, which contains tools related to identification, design and control experimentation systems. Hough et al. [2002] proposed an interactive learning website, where students can learn at their own schedule and pace under any of these three schemes of interaction: *interactive learning*, *tutorial* and *instrumentation notes*.

^{*} This work was supported by the *Autotronics* research chair at *Tecnológico de Monterrey, Campus Monterrey*.

Martin et al. [2005] presented a methodology that combined both modelling and interactive simulation. The methodology combines the use of *Ejs*, *Matlab* and *Dymola* (with *Modelica*), which implies object-oriented design and programming of *JARA*.

Alves et al. [2006] developed an *EDU* educational *SCAda* (*EDUSCA*), a tool that incorporates *OPC* server and that can be integrated with experimental systems to develop practical activities. Students can combine theoretical knowledge with practical experience.

Lim [2006] proposed a method, in which students write control software using a programming environment based on the real-time operating system *VxWorks* from *WindRiver Systems*. Here, an analog dynamic simulator represents the process.

Xuejun et al. [2007] implemented a real-time digital simulation that requires no hardware. The *LabVIEW* Simulation Interface Toolkit provides an effective integration between *MATLAB/Simulink* software and *LabVIEW*.

2.2 Laboratories

Other researchers have considered real applications. Examples are cited next. Lim et al. [1991] created a low cost workstation prototype consisting on a personal computer with analog input-output adapters and a software package based on *Matlab*. This tool monitors a pair of coupled water-tanks and/or motor sets. Etxebarria et al. [1998] worked with a system, in which a digital signal processor receives firmware generated in *Simulink* and in *Matlab RealTime Workshop RTWorkshop* and then runs autonomously.

Bequette and Ogunnaike [2001] emphasized the distinctly challenging characteristics of chemical processes. A review of how different universities had been working with is included. A control studio that combines lectures, simulations, and experiments in a single classroom is presented.

Ang and Braatz [2002] proposed a course with an emphasis on project-based learning, where students gain experience by working with physical processes. A modern control interface was implemented in *HP-VEE*, which is a modern visual programming environment for instrument control. This interface can communicate with 11 processes.

Bachnak and Steidley [2002] described a series of acquisition systems based on laboratory data that rely on the National Instrument (*NI*) systems and *Matlab*. It included flow processes and air heating systems. Saco et al. [2002] developed a similar work.

Klan et al. [2005] developed *CTRL*, a portable, autonomous microcontroller-based interface, which facilitates real-time control in *Matlab*. The application was tested with a set of experimental models: hydro-pneumatic, rotation speed-control, ball and ellipse, air temperature, and time delay.

Novak et al. [2006] designed a control laboratory which integrates the skills from theoretical courses to solve engineering problems in dynamic and control systems. It was designed from eleven experiments based on commercial/educational equipment, and *Matlab/Simulink*.

2.3 Remote laboratories

Remote laboratories are mainly used academically to enhance classroom lectures, share research equipment, and complement the learning process. Currently, the majority of all available remote experiments and laboratories come from the area of measurement, while the rest cover automatic controls. This approach simplifies the logistics and covers the needs involved in conventional laboratory work, including scheduling of equipment, space, staffing, training, and safety, among others. Some examples are presented next.

Zhang et al. [2004] introduced *Netlab*, a remote laboratory that interacts with real devices, such as DC systems, inverted pendulum, triple tank systems, etc. It can execute *ANSI C* compiled scripts in the form of *DLLs*. Burchett and Layton [2005] proposed a similar work

Casini et al. [2004] presented a remote laboratory, based on *Matlab/Simulink*. It enables user to select the control algorithm online. Valera et al. [2005] also created an interactive laboratory: via *WinCon*. This remote laboratory interacts with a servomotor and shaker table. Coito et al. [2005] and Selmer et al. [2005] proposed similar approaches and developed a *LabView* remote control solution.

Davari and Chandrasekara [2005] worked with a torsional and a rectilinear control system in a laboratory, and created a software executive program¹.

Viedma et al. [2005] presented *iLab*, a system based on the *TCP/IP* protocol. A computer running any operative system can be connected to a server that provides analysis and design tools.

e-learning systems are often limited by internet traffic. Some researchers have proposed models which overcome this constraint. Chaabene et al. [2006] proposed a system consisting on a set of software elements embedded in the local control that get over internet traffic.

Wu et al. [2006] presented an internet-based control-engineering laboratory that provides remote access to many experimental systems, where students can choose or design a controller. Online features include real-time experimentation with a fan/plate, coupled tank, DC servo, etc.

Hassan et al. [2007] developed *simPROces*, a tool that allows remote development and validation of embedded control applications. A pilot water tank is the example base case. Gao and Lin [2007] also developed an Internet-based water level control system.

Hercog et al. [2007] introduced a framework in which *Matlab/Simulink* was used for control algorithm development, while *LabView* was used for the user front-end and remote control. Remote users easily interact with a set of physical control experiments.

Uran and Jezernik [2008] proposed a computer-aided learning strategy. On it, a virtual web-based laboratory working on *MATLAB Web server* supports *CAD* and structured experiments. Students are allowed to create and execute experiments online.

¹ www.ecpsystems.com

3. ACADEMIC PROPOSAL

Two main directions are taken into account for the design of an engineering course at *Tecnológico de Monterrey*: the *2015 Mission* and the *Accreditation Board of Engineering and Technology (ABET²)* criteria.

On this line, an educational model, which has been characterized by its richness on information technologies (i.e. *Blackboard Academic Suite³*) and the systematic incorporation of teaching techniques, was developed at *Tecnológico de Monterrey*. Students assume an active role in their learning process and build knowledge on the basis of their own experience, under the direction and guidance of their professors, Mijares [2007].

Based on the model, a new control engineering course was designed, giving special attention to experimental teaching. As several constraints were identified in previous projects Morales-Menendez et al. [2006], a research plan was started in 2006 looking for a solution for some limitations. The goals of this research require that students:

- (1) *interact* with industrial equipment (i.e. sensors, actuators, controllers, etc) .
- (2) *avoid* the time consumed in the installation and configuration steps.
- (3) *design and implement* different control strategies

Around 1,000 students at *Tecnológico de Monterrey* work every year with only few tank-level control stations (Fig. 1). Therefore, a software tool that communicates a personal computer with a *Honeywell UDC 6300* controller was developed.

A tank-level monitoring and control system represents the dynamics of several processes. The instruments installed on this station (Fig. 1) are industrial devices, such as: 2 flow transmitters (FT-101, FT-102), 1 level transmitter (LT-00), and 1 control valve (FV-100). The *PID* controller is a *Honeywell UDC 6300* with a serial communication module. The characteristics of this level-tank control station met the first aforementioned goal of this research project: interaction with industrial equipment.

However, two more challenges were still to be overcome: 1) the development of a *communication system* between the controller and the computer, and (2) the facilitated integration of the *control system application*.

3.1 Communication system.

There exist multiple brands of devices, protocols and industrial network standards in the automation field. At the beginning, the costs of software maintenance increased as users tried to interconnect with devices from different manufacturers, while suppliers developed their own connectivity standards.

In 1996 several process automation company leaders joined (*Fisher-Rosemount, Rockwell Software, Opto 22, Intellution and Intuitive Technology*) with the goal of creating the first common interface specification and roadmap. Since most companies and customers use *Microsoft Windows*,

² www.abet.org

³ www.blackboard.com

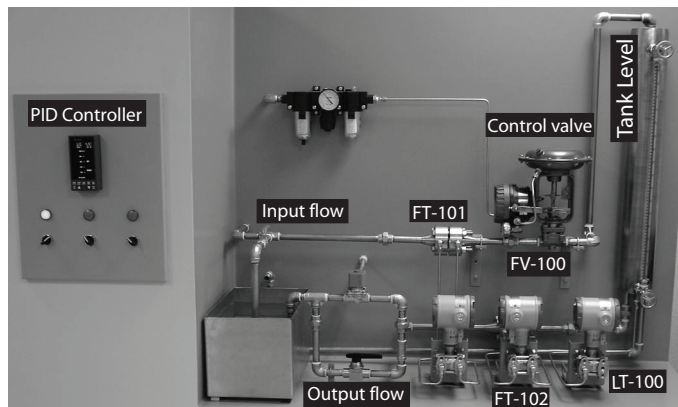


Fig. 1. Tank-Level monitoring and control station.

the joint venture decided to include Microsoft Corporation in their initiative. As a result, they created the *Object Linking and Embedding (OLE) for Process Control (OPC)* foundation. *OPC* is a set of *Application Program Interfaces (APIs)*, based entirely on *COM* objects, that move data from one process to another, without creating a new implementation for each product. At the end, each manufacturer embeds a set of common interfaces to exchange data between processes using the same interface, with common structures and methods.

One of the most used techniques for data exchange is windows sockets (*Winsock*). It is a specification for network applications, precisely *TCP/IP*-based ones. This tool defines a standard interface between a *TCP/IP Client* and a *TCP/IP Server* and is one of the most important building blocks on operative systems. Its name comes from the *UNIX Berkeley sockets*, the first data exchange implementation. In 1993, the first version of *Winsock* was launched .

In some applications, it is possible to implement a *loopback* connection. The *loopback* address for any computer is defined by the *IETF* in memo RFC3330 as 127.0.0.1. Using this approach, one application is configured as the server and the other one as a customer. After that, both can exchange data by writing to a virtual high-speed file.

Fig. 2 shows the architecture of the communication system. The software tool has a special module designed to monitor and manage communication with the controller. This process is done through a *RS485 to RS232* converter/modem connected to a personal computer. The main challenge was the interaction *Matlab/Simulink* and *ANSI C* on soft real-time.

There are two additional interface modules. One interface is responsible to send and receive data from the software running on *Matlab/Simulink*. The communication is done through a *loopback TCP/IP* connection on the personal computer. Once the simulation is started on *Matlab/Simulink*, the time is controlled by the software.

3.2 Control system applications.

Taking into account the developed interface modules, it is possible to use any of the toolboxes that run in *Matlab/Simulink*; the only restriction is that these toolboxes can be used only with a discrete solver. The *Mat-*

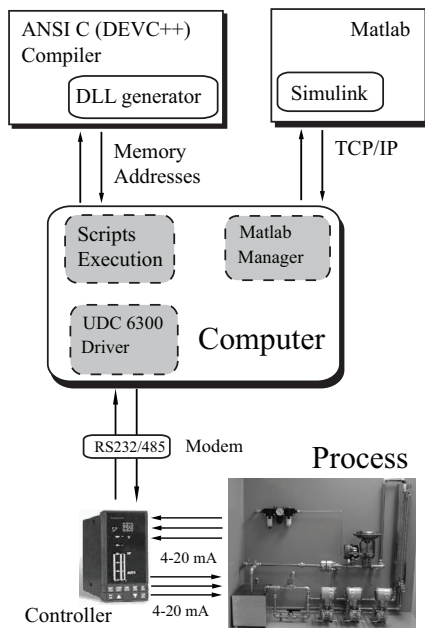


Fig. 2. *Communication system*. The software tool is linked to the controller by a serial system with a *RS232/485* modem. The controller has a standard 4-20 mA communication with sensors and actuators. By *TCP/IP* the software tool exchanges data with *Matlab/Simulink* software, and through the memory, it communicates with *ANSI C* scripts.

lab/Simulink toolboxes represent a great feature as they work with many types of applications.

Matlab/Simulink facilitates model creation by offering block diagram tools, as well as an easy and flexible environment for implementation of several control strategies. The software also contains a set of libraries for predefined tasks.

A model containing the variables of the process is provided for students, Fig. 3 (left picture). Users need only to plug and play with this model. By these means, the second aforementioned goal is fulfilled: *avoiding* the time-consuming configuration step.

Moreover, *ANSI C* libraries (known as *DLL*) can be compiled and used to implement many possible algorithms. The complete coding framework is provided in order to produce the compiled controller. Students will only be focusing on the design of the algorithms, as the software tool hides all the issues related to the communication system.

4. RESULTS

The software tool was tested during two semesters in a senior automation control-systems course. In the first term, three teams (4-students) worked with it. Three different projects were implemented: (1) a fuzzy logic controller Lopez et al. [2008], (2) cascade and feed-forward control strategies Morales-Menendez et al. [2008], and (3) a feedback control strategy with different *PID* tuning techniques.

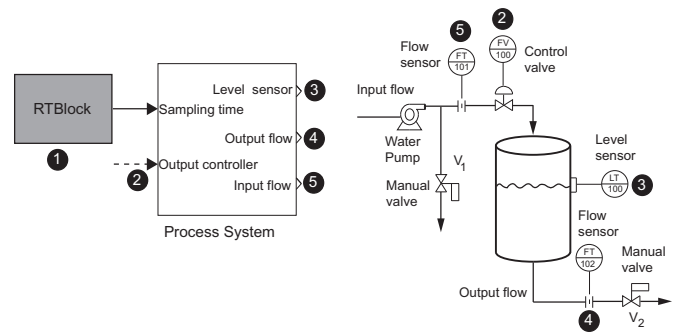


Fig. 3. Tank-level *Simulink* model. Using the left model, students write the (2) output controller and read the process variables: (3) level, (4) output flow and (5) input flow of the tank-level control station. Using the *RTBlock*, the sampling time can be defined. The numbers show the virtual and physical equivalences between the level-tank station diagram shown on the right and the *Simulink* model.

In the second term, two new strategies were defined, designed and implemented: (1) a *Smith predictor* control and (2) an *Internal Model Control (IMC)*. Due to space restrictions in this paper, only the *IMC* project is presented next.

The right picture in Fig. 3 shows a basic instrumentation diagram of the tank-level control station. An *IMC* controller was implemented (Fig. 4) in order to keep the level-tank (LT-100) at the set point value. This was achieved by manipulating the input flow (FT-101) through the control valve (FV-100).

Testing was done in order to evaluate the performance of the system. Plots of the set point change and regulator problem are shown in Fig. 5. The results show that the *IMC* controller has a better performance than a standard *PID*.

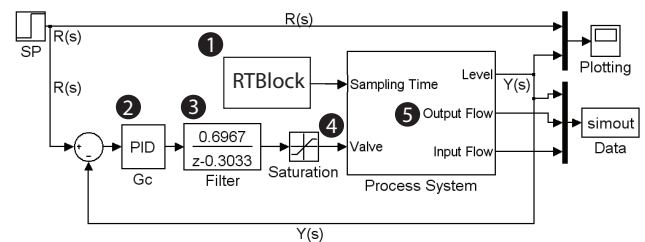


Fig. 4. *IMC* controller model based on *Matlab/Simulink*.

During experimental work, students become acquainted with real-life features, gaining experience and knowledge that would otherwise not be obtained. Students showed high motivation while working in the project.

5. COMPARISON WITH COMMERCIAL SOLUTIONS

This research project started with the three main goals established in the academic proposal. As, the economic aspect is also crucial, review of several commercial applications is considered in addition to the state of the art presented in section 2.

It is important to emphasize that the comparison made in this section is in the context of this specific application (tank-level monitoring and control stations), where the

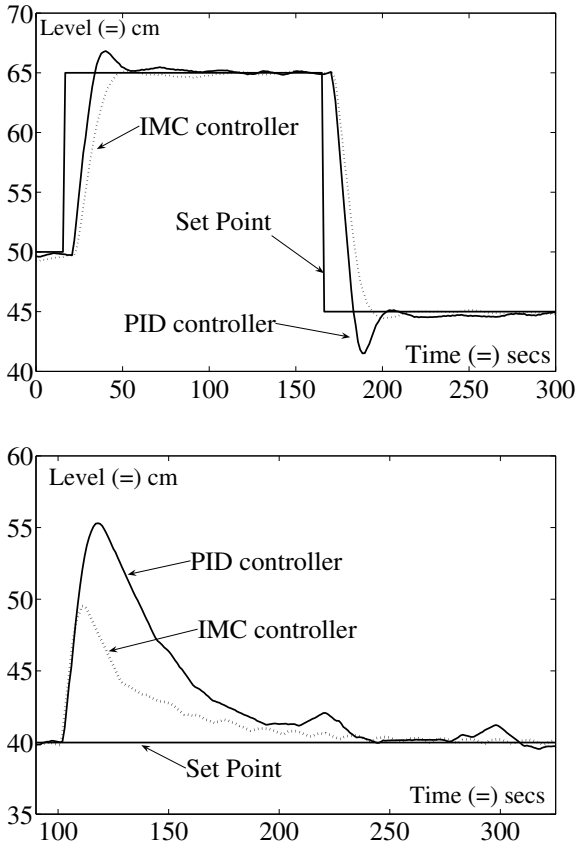


Fig. 5. *IMC* controller performance. Comparison between the *PID* controller and the *IMC* controller for a servo control approach (top) and regulator problem (bottom).

academic point of view is mandatory. Considering the fact that students have limited time for each course, and that universities have limited budget for their academic laboratories, training and operating costs are also essential parts of this analysis.

*RSView*⁴ is a tool designed and commercialized by *Rockwell Automation* that allows customers to easily human-machine interfaces as well as applications with acceptable real-time functionality. It also offers connectivity with *TCP/IP*, *RS232* and *OPC* drivers from several manufacturers. On the down side, a lot of training for users is required, and it is a high cost solution.

Matlab offers support for some of the *National Instruments (NI)*, *AdvanTech* and *Measurement Computing* acquisition cards, and for *OPC*. It contains an impressive analysis library, widely used by universities and research centers. However, its cost is extremely high as toolboxes are incorporated to the suite. This proposal includes the incorporation of the *Matlab* toolboxes.

LabView and *LabWindows/CVI*, are *ANSI C*-based programming tools for test, measurement and control, designed and commercialized by *NI*. This software includes drivers for the entire catalog of acquisition cards from *NI*, and also offers support for different standards such as *VXI*,

⁴ www.rockwellautomation.com

PXI, *VISA*, *GPIC*, *OPC*, *Field Point*, etc. It is possible to create programs with acceptable functionality in real time, in this low cost tool; however, it requires a lot of training.

*Siemens' SIMATIC WinCC*⁵ a tool that allows users to create a human-machine-user interface easily, offers connectivity with *PPI*, *MPI*, *PROFIBUS-DP*, *TCP/IP*, *RS232* and drivers for *OPC* from several manufacturers. It has the same restrictions that *LabView* and *LabWindows/CVI* have.

*Agilent VEE*⁶ is a low cost development language for test, measurement and control. The programming is based on the "drag a drop" of blocks in the interface. It offers support for different standards such as *VXI*, *PXI*, *VISA*, *GPIC*, *OPC*, *Field Point*, etc. It is possible to create programs with acceptable functionality in real time, however it requires a lot of training to users.

*Honeywell's Profit Suite 300*⁷ is a tool that allows users to create a human-machine-user interface easily, and also offers connectivity with *TCP/IP*, *RS232* and drivers for *OPC* from several manufacturers. However, it has the same constraints as *Agilent VEE*.

6. CONCLUSIONS

This academic proposal can help students to develop and operate real control systems. By the elements presented in this paper, the teaching/learning process for students with no background in electrical or computer sciences can be facilitated, as hands-on laboratory sessions eliminate the gap between theory and practice.

Several skills and abilities are promoted as part of the teaching system. Starting from a real problem, students can find the solution in a framework for significant learning, developing concrete abilities. With this, theoretical concepts can be better understood, facilitating knowledge transference to other contexts.

Limitations. The minimum sampling time is an important constraint, as it is serial communication (*RS232-RS422/485*). In addition, a discrete solver is needed. However, in the majority of the cases this problem can be solved, since the developed communication system applies only for the *Honeywell UDC 6300* controller.

Future work. Reduction of sampling time is mandatory. The software tool can be improved with an *OPC* interface. This will increase the number of devices to communicate with in a more reliable way.

ACKNOWLEDGEMENT.

The authors would like to thank: A.Espino, A. Castañeda, V. Mansilla García, M.Delgado, F. Zambrano, and R. Villegas, for sharing their experimental results.

REFERENCES

R. Alves, J.E Normey, A. Merino, and C. de Prada. EDUSCA (EDUCational SCAda): Features and Applications. In 7th *Adv. in Control Education*, Spain, 2006.

⁵ www.automation.siemens.com

⁶ www.home.agilent.com

⁷ www.honeywell.com/ps

- S. Ang and R.D. Braatz. Experimental Projects for the Process Control Laboratory. *Chem. Eng. Edu.*, 36:182–187, 2002.
- R. Bachnak and C. Steidley. An Interdisciplinary Laboratory for Computer Science and Engineering Technology. *J. Comput. Small Coll.*, 17(5):186–192, 2002.
- B.W. Bequette and B.A. Ogunnaike. Chemical Process Control Education and Practice. *IEEE Control Systems Magazine*, pages 10–17, 2001.
- B.T. Burchett and R.A. Layton. An Undergraduate System Identification Experiment. In *American Control Conf*, volume 7, pages 5145–5149, USA, June 8-10 2005.
- M. Casini, D. Prattichizzo, and A. Vicino. The Automatic Control TeleLab – A Web-based Technology for Distance Learning. *IEEE Control Syst. Mag.*, 24(3):36–44, 2004.
- M. Chaabene, K. Mkaouar, and R. Souissi. A Web-based Control of a Real Laboratory for Process Engineering Education. In *World Automation Congress*, Hungry, July 2006.
- F. Coito, P. Almeida, and L.B. Palma. SMC RVI - A Labview/Matlab based Tool for Remote Monitoring and Control. In *10th ETTA*, volume 2, pages 1039–1044, Sept 2005.
- D.J. Cooper and D. Dougherty. Enhancing Process Control Education with the Control Station Training Simulator. *Computer Applications in Engineering Education*, (7):203–212, 1999.
- R.K. Cox, J.F. Smith, and Y. Dimitratos. Can Simulation Technology Enable a Paradigm Shift in Process Control? Modeling for the rest of us. *Computers and Chemical Engineering*, 30:1542–1552, 2006.
- A. Davari and C. Chandrasekara. Control Experimentation for Undergraduate Student. In *American Control Conference*, volume 7, pages 5156–5161, Portland, OR, June 8-10 2005.
- V. Etxebarria, I. Lizarraga, and J.M. Alcalde. An Integrated Environment for Practical Control Engineering Courses. *Int. J. of Electrical Eng. Education*, 35(1):3–13, 1998.
- L. Gao and J. Lin. Labview and Internet Based Remote Water Level Control Laboratory. pages 187–188. IEEE, 2007.
- H. Hassan, C. Dominguez, J.M. Martinez, and J. Albaladejo. Remote Laboratory Architecture for the Validation of Industrial Applications Control. *IEEE Trans on Industrial Electronics*, 54(6):3094–3102, 2007.
- D. Hercog, B. Gergič, S. Uran, and K. Jezernik. A DSP-Based Remote Control Laboratory. *IEEE Trans on Industrial Electronics*, 54(6):3057–3068, 2007.
- M. Hough, E. Word, S. Yip, and T. Marlin. A Web Site to Support Active Student Learning in Process Control. In *ASCE Annual Conference & Exposition*, 2002.
- R. Irawan, M. Ooi, G. Yeung, E. Weyer, D. Nestic, and I. I. Mareels. A Virtual Laboratory Experience based on a Double Tank Apparatus. In *40th IEEE Conf. on Decision and Control*, volume 3, pages 2815–2820. IEEE, 2001.
- S.L. Jämsä-Jounela. Future Trends in Process Automation. *Annual Reviews in Control*, pages 211–220, 2007.
- P. Klan, M. Hoefreiter, J. Machacek, O. Modrlak, L. Smutny, and V. Vacek. Process Models for a New Control Education Laboratory. In *16th IFAC World Congress*, pages 139–144, Prague, Czech Republic, 2005.
- D.J. Lim. A Laboratory Course in Real-Time Software for the Control of Dynamic Systems. *IEEE Transactions on Education*, 49(3):346–354, 2006.
- K.W. Lim, C.T. Ong, and E. Sivagnanam. Development of a Low Cost Control Workstation. In *IECON '91*, pages 1375 – 1379, Nov 1991.
- T. Lopez, R. Morales-Menendez, R. Ramirez, and F. Guedea. A Human Machine Interface for Teaching Continuous Control Systems. In *Modelling, Identification and Control*, pages 166–171, Innsbruck, Austria, Feb 2008.
- C. Martin, A. Urquia, and S. Dormido. Object-Oriented Modeling of Virtual Laboratories for Control Education, 2005.
- C. Mijares. PDHD Teaching Skills Development Program. Technical report, ITESM system, 2007.
- R. Morales-Menendez, I.Y. Sanchez-Chavez, M. Ramirez, and L.E. Garza. Control Engineering Education at Monterrey Tech. In *American Control Conference*, pages 286–291, Minneapolis, USA, June 2006.
- R. Morales-Menendez, T. Lopez, R. Ramirez, and L.E. Garza. Drag-and-Drop Graphical User Interface for Process Control Education. In *to appear in ASCE Annual Conference & Exposition*, Pittsburgh, USA, 2008.
- J. Novak, P. Chalupa, and V. Bobal. New Laboratory Course For Control Education. In *7th Advances in Control Education*, number 1, Madrid, Sp, 2006.
- R. Saco, E. Pires, and C. Godfrid. Real Time Controlled Laboratory Plant for Control Education. In *32nd ASCE/IEEE Annual Frontiers In Education*, volume 1, pages 12–16, Boston, Mass, Nov 2002.
- N. Schroeder. Process Automation Markets in 2010. Technical report, Intechno Consulting, 2003.
- A. Selmer, M. Goodson, M. Kraft, S. Sen, V.F. McNeill, B.S. Johnston, and C.K. Colton. Performing Process Control Experiments Across the Atlantic. *Chemical Eng. Education*, pages 232–237, 2005.
- S. Uran and K. Jezernik. Virtual Laboratory for Creative Control Design Experiments. *IEEE Transactions on Education*, 51(1):69–75, 2008.
- A. Valera, J.L. Diez, M. Valles, and P. Albertos. Virtual and Remote Control Laboratory Development. *Control Systems Magazine*, 25(1):35–39, 2005.
- G. Viedma, I.J. Dancy, and K.H. Lundberg. A Web-based Linear-Systems iLab. In *American Control Conference*, volume 7, pages 5139–5144, Portland, OR, June 8-10 2005.
- H. Wu, Y. Yang, W. Qingquan, and S. Zhu. An Internet-based Control Engineering Laboratory for Undergraduate and Graduate Education. In *IEEE Int. Conf. on Automation Science and Eng.*, pages 546–550. IEEE, October 2006.
- X. Xuejun, X. Ping, Y. Sheng, and L. Ping. Real-time Digital Simulation of Control System with LabVIEW Simulation Interface Toolkit. In *26th Chinese Control Conf*, pages 318–322, China, July 26-31 2007.
- S. Zhang, H. Wu, Q. Lin, and S. Zhu. Netlab-based Control Experiments for Control Engineering Education. In *8th Control, Automation, Robotics and Vision Conference*, volume 3, pages 2188–2193, Dec 2004.