MECHATRONICS/PROCESS CONTROL REMOTE LABORATORY

Hong Wong, Vikram Kapila, and Anthony Tzes

Department of Mechanical Engineering
Polytechnic University, Brooklyn, NY

Abstract

Under an NSF—DUE sponsored laboratory development program, we have developed a multidisciplinary mechatronics/process control remote laboratory (MPCRL) consisting of an array of experiments, which expose students to elements of aerospace, mechanical, electrical, civil, and chemical engineering. A new laboratory curriculum and manual have been developed to introduce students to PC-based data acquisition, rapid control prototyping, and control of a multitude of multidisciplinary experimental test-beds. In addition, in summer 2000, we developed the MPCRL web site to facilitate remote access to our laboratory test-beds via the world-wide-web. The MPCRL web site features online-experiments, information/navigation/ resource centers, prerecorded videos of experiments, live video stream of online-experiments, and a chat window. The MPCRL supports undergraduate and graduate control courses including the capstone design projects. Finally, its outreach efforts have included summer workshops for graduate and high school students.

Introduction

Engineering education is facing unprecedented challenges and exciting opportunities. Advances in communication and information technology are reshaping our society in unparalleled ways. With the ever frequent reengineering and restructuring of corporate America, engineers are often working in disciplines that transcend their formal education. For example, many engineering graduates follow diverse career paths in medicine, financial engineering, management, public policy, and other fields that require a systems-oriented analysis and synthesis ability. In addition, the highly complex and multidisciplinary nature of modern engineering systems demand synergies between various engineering and science disciplines and a strong synthesizing approach. These dynamics point to the acute need for training engineering students in a multidisciplinary, cooperative, active-learning environment.

An integral part of modern engineering systems is the design of advanced control strategies, which require interplay across disciplinary boundaries, necessitating a thorough understanding of fundamentals of multiple engineering disciplines in order to analyze and synthesize emerging technological products. For example, breakthrough control design concepts for hybrid fuel cells, intelligent highways, space-plane, biologically inspired systems, earthquake-proof civil structures, automated drug delivery systems, etc., demand a clear understanding of the underlying physical laws governing these systems; a multidisciplinary venture in itself. In light of the above, it is of paramount importance to develop an experimental control laboratory curriculum that allows students to conduct design, analysis, implementation, and validation of controllers for a broad array of multidisciplinary test-beds, thus enabling them to become well-rounded engineers who can meaningfully contribute to the advancement of the engineering society.

Finally, recent years have witnessed a phenomenological growth of the Internet and the world-wide-web (WWW). Diverse array of governmental and commercial activities are now conducted online. Within the education sector, WWW is being exploited for rapid information dissemination and content-rich delivery of educational material to enhance student learning and to revolutionize distance learning. In this paper, we describe our efforts toward the development of a Mechatronics/Process Control Remote Laboratory[1] (MPCRL) that is designed to provide students *i)* a multidisciplinary experimental control experience and *ii)* the ability to access and interact with selected laboratory experiments using WWW beyond the laboratory hours; thus enhancing their overall laboratory experience.

Objectives

To respond to the aforementioned need for a multidisciplinary control laboratory curriculum, we have established a pilot program, *viz*., the MPCRL. The objectives of this program have been:

- Develop a multidisciplinary laboratory to expose students to the full spectrum of instrumentation and real-time control.
- Strengthen students' comprehension of abstract concepts through experimental validation.
- Enable students to benefit from collaborative and active learning projects involving the design and development of experimental control projects from the conceptualization to the final prototype stage.
- Enrich students' educational experience by bringing real-world projects to engineering education.
- Implement an Internet-based remote access laboratory paradigm to enhance student learning.

Program Overview

To address the above goals, we have focused on developing a laboratory where principal experiences in experimental methodology include elements of modeling (dynamics, heat transfer, fluid mechanics, process reaction kinetics, structural vibration, etc.), hardware integration (plant, sensor, actuator, power electronics, and data acquisition), and a unified software environment for control system design, analysis, implementation, and validation. Specifically, for a given experimental test-bed, a rigorous development of its model using first principles of the underlying physics enables students to understand its mathematical/physical characteristics (e.g., frequency/time responses, resonant frequencies, etc.). Furthermore, the integration of plant and sensor allows students to interface transducers, such as microphones, encoders, accelerometers, pulse flow transmitters, etc., with power electronics (filters, converters, pre-amplifiers, etc.). Similarly, the integration of plant and actuator enables students to interface actuators, such as speakers, motors, pumps, servo valves, electromagnets, etc., with power amplifiers. Next, using system identification techniques on experimental test-beds, students learn the procedure of validating mathematical models *vis-à-vis* experimental data. Finally, the process of control design, analysis, implementation, and evaluation provides an active method to improve operational characteristics of a system. These modeling, hardware interface, system identification, and control design exercises expose students to salient real-world issues in control design and implementation such as sensor noise, actuator saturation, amplifier sizing, notion of control objectives, control algorithm implementation, sample-rate selection, etc.

The MPCRL has been focused on providing students access to a wide variety of real-world, multidisciplinary control test-beds (see Ref. 2 for a complete listing) and an abundant resource of information and laboratory time. The MPCRL is an innovation designed to enable students to work on laboratory experiments beyond the traditional three hours per week, through the Internet. It consists of *i)* a web site that connects students to online-experiments controlled and monitored through the Internet, *ii)* detailed specifications and operation manuals of various experiments, *iii)* prerecorded videos of selected experiments, *iv)* live streaming video of online-experiments, *v)* a chat window to communicate with other students and to form Internet-based remote learning communities, *vi)* email addresses of professors and teaching assistants, and *vii)* links to miscellaneous control related web sites. These features of the MPCRL provide students with a wealth of information and virtually unlimited remote access to the laboratory test-beds to motivate them to take an active role in their education.

Beyond providing a real-time experimental control experience to undergraduates, an array of additional educational/research activities have been generated under this effort.[3,4,5] In particular, the modular nature of the experimental hardware is being exploited to support upper-level undergraduate and graduate courses. Furthermore, the seamless integration and operation of experimental hardware and control design, analysis, implementation, and validation software, described below, make this real-time environment ideal for undergraduate summer research, capstone design, and graduate student research; enabling students to explore advanced control strategies on a multitude of real-world experimental test-beds. Finally, even in its preliminary stage, the MPCRL has contributed to outreach programs including the Youth in Engineering and Science (YES) program,[4] which encourages high school students to pursue studies in engineering and science, and a Summer Workshop[5] to expose graduate students from various universities to real-time experimental control. This array of activities facilitates the development of competencies in project-based, cooperative, and peer learning (e.g., through hands-on activities, team projects, group discussions, and team writing) and active learning (e.g., thinking, observing, brain-storming, listening, note taking, critical reading, summarizing, problem solving, conducting computer simulation, doing technical research, self-learning, making oral presentation, etc.) among students. Furthermore, this program exposes students to the excitement of discovery.

Background

As part of the undergraduate laboratory course sequence in the control and robotics area, until the academic year 1996, the mechanical engineering department used to offer a combined instrumentation and control laboratory. In 1996, instrumentation (ME 324) and control (ME 325) were established as two separate laboratories and offered concurrently every spring semester. The revised instrumentation laboratory curriculum focused primarily on sensors, actuators, data acquisition, and LabVIEW. The control laboratory provided a preliminary coverage of direct current (DC) motor, flexible beam system identification, and some proportional-integral control. An in-depth, hands-on treatment of feedback control, hardware/software integration, etc., was relegated to the senior design sequence ME 361 and ME 362, which provided a strong emphasis in mechatronics.

In a parallel development, starting fall 1996, we began developing a real-time control laboratory for graduate research and education. This laboratory development effort exposed our graduate students to modern concepts of digital signal processor and PC-based real-time control, automated C-code generation, etc. Our initial experience in pilot-testing the above real-time controller implementation environment with the graduate student population was extremely beneficial as it trained our graduate students to later assume the role of teaching assistants. Having successfully developed the graduate real-time control laboratory, we took the next natural step toward the modernization of the undergraduate control laboratory. Specifically, starting in spring 1999, we began developing a real-time experimental control laboratory which aims to provide our undergraduates a comprehensive hands-on exposure to the concepts of real-time data acquisition, rapid control prototyping, system identification, classical/modern feedback control, etc., on a variety of multidisciplinary experimental test-beds.[3]

In the sequel, we provide a brief description of the experimental hardware configuration and a standardized experimental methodology followed in the development of the MPCRL. Some safety protocols in designing online-experiments are discussed. Finally, student evaluations and comments assessing the efficacy of the MPCRL are presented.

Experimental Hardware Configuration

Hardware is the backbone that shapes an evolving control laboratory. A control laboratory promotes experimental learning as a complement to theoretical learning. As students interact with experiments, they familiarize themselves with current technology. The MPCRL provides an experimental hardware environment, shown in Figure 1, which consists of many multidisciplinary test-beds, power amplifiers, MultiQ-3 data acquisition and control boards[6] (DACB), Internet cameras, web-enabled client PCs, and web server PCs. The MultiQ-3 DACB is effortlessly interfaced with experimental test-beds and power amplifiers obtained from diverse sources. In addition, the MultiQ-3 DACB is accessible through a unified software environment described below, thus providing a platform that enables rapid control design, analysis, implementation, and validation.

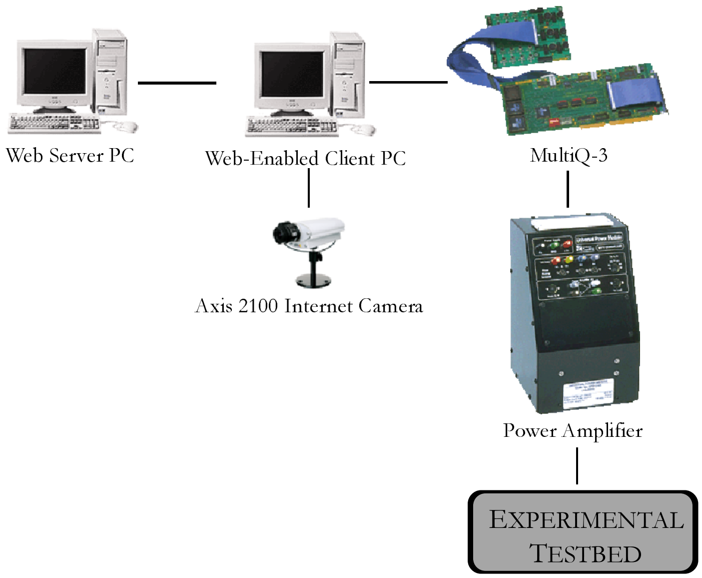

Figure 1: Experimental hardware environment

- Multidisciplinary experimental test-beds currently available in the MPCRL include electro-mechanical systems (DC servomotors, magnetic levitation systems, inverted pendulums, aerial vehicles, mobile robots), thermo-fluid systems (level, flow, temperature, and pressure control), acoustics (active noise cancellation), structural systems (rotary flexible link, earthquake simulator), and chemical processes (pH control). The governing principles describing the dynamics of these systems include laws of Newton, Kirchoff, electromagnetism, mass/energy conservation, etc. In addition, these experiments utilize a diverse array of sensors and actuators for process monitoring and feedback control. A representative list of sensors and actuators employed in these experiments is given in Tables 1 and 2, respectively. Further details on principles of operation of various experiments can be found in Refs. 7—9.
- The client PC, with Windows NT 4.0 operating system and VenturCom RTX real-time kernel,[10] provides a platform to rapidly implement and validate complex control architectures for real-time feedback control of experimental test-beds. Furthermore, the client PC's are web-enabled thus providing a gateway for remote users to access online-experiments.
- MultiQ-3 DACB provides the following functionalities: analog to digital conversion (ADC), digital to analog conversion (DAC), digital I/O, and encoder readout. Through the MultiQ-3 DACB a web-enabled client PC monitors and controls an experimental test-bed.
- Power amplifier modules perform two functions: supply power to sensors (including data pre-filtering) and amplify DAC outputs to drive actuation elements.
- A web server PC, with Windows NT 4.0 operating system, retains all the web pages for the MPCRL. Using Apache server,[11] a freely available open source code implementation of a web server, the web server PC facilitates viewing of the MPCRL web site and access to its diverse features mentioned previously, through the WWW.

- Axis 2100 Internet cameras[12] provide live streaming videos of the online-experiments. These cameras connect directly to the WWW through the Transmission Control Protocol/Internet Protocol (TCP/IP). Using the Axis 2100 Internet camera's hardware-embedded Internet streaming video software, remote users can receive live streaming video of online-experiments in action.

Table 1: List of sensors used in experimental test-beds

Sensors	Description
Potentiometer	Angular/translational position is measured using a variable resistance
Tachogenerator	Angular velocity is measured using an electric generator
Accelerometer	Acceleration is measured through displacements of a referenced mass
Incremental Encoder	Angular position is measured using a set of pulse train signals to indicate number of counts and direction of motion
Limit switches	Absolute reference position is measured using a light emitting source
Light camera sensor (based on silicon photodarlington)	Linear displacement is measured using phototransistors to sense variations of light intensity from a moving light source
Float level	Level of liquid in a tank is measured using a float driven potentiometer
Flow rate	Liquid flow rate past a rotating impeller is obtained by measuring an impeller's frequency
Omni-directional microphones	Acoustic pressure at any incidence angle is measured through displacements of a diaphragm
pH probe	Level of pH is measured using a pH sensitive membrane
Pressure sensor	Pressure is measured using a set of active element piezoresistive Wheatstone bridge
Thermistor	Temperature is measured using a variable resistance

Table 2: List of actuators used in experimental test-beds

Actuators	Description
Servomotor	Provides torque from voltage signals
Servo-controlled fans	Provides thrust from rotation of fans controlled by servomotors
Servo valve	Controls flow-rate using a gate (controlled by a servomotor) to block/open the path of the liquid
Pneumatic control valve	Controls pressure at a point using a diaphragm actuator
Solenoid valve	Switches liquid flow through a pipe on/off by passing current through a solenoid coil
Electromagnet	Provides electromagnetic force from currents signals
Cone speaker	Induces a pressure field from alternating voltage signals
Variable flow-rate pump	Supplies liquid at desired flow-rate based on voltage signals
Fixed voltage centrifugal pump	Supplies a fixed flow-rate into a piping system using a direct current motor driving a rotor
Heat exchanger	Transfers heat between fluids at their separation boundary

Standardized Experimental Methodology

Experimental methodology is the muscle that facilitates the attainment of the aforementioned goals of the MPCRL by providing students a streamlined architecture to perform various aspects of hands-on, active learning. To fulfill the objectives of the MPCRL, students perform calibrations on experimental test-beds, both in hardware and in software. They become familiar with the software environment that uses modern control analysis, design, and implementation tools like Matlab,[13] Simulink,[13] Real-Time Workshop[13] (RTW), WinLib,[14] and WinCon.[14] Finally, web-enabled experiments are enhancements to the MPCRL that provide a student-centric, flexible learning environment wherein self-paced learning is accomplished.

A. Calibration

Students usually disregard calibration as a mundane task and consider it as a small component in the entirety of a laboratory experiment. It should be emphasized that by performing calibration, fundamentals of sensors, actuators, and their integration with power electronics can be understood. Furthermore, a properly calibrated experimental test-bed eliminates potential closed-loop instability arising from erroneous read-out of physical measurement and/or erroneous application of control effort. Finally, for students desirous of designing high-performance controllers, detailed knowledge of experimental test-bed subsystems helps them design controllers that can tweak a system's performance to the maximum!

For each in-class experiment, students are required to calibrate sensors and actuators. These preliminary procedures are performed both in software and in hardware. The software calibration uses properly designed filter/gain blocks, within a Simulink block-diagram architecture, to transform the sensory information acquired from the ADC to corresponding physical quantities (motion, pressure, level, etc.). In addition, the hardware calibration allows students to eliminate sensor output bias and to utilize full-scale of the ADC by controlling the offset and gain knobs, respectively, on sensor electronics.

B. Primary Software Environment

On a web-enabled client PC, the primary software environment enables rapid control prototyping via Matlab, Simulink, RTW, WinLib, and WinCon software. Figure 2 shows the hierarchy of the primary software environment. Matlab is at the heart of the primary software environment. The types of functions that Matlab can perform are too numerous to list; in our context, Matlab facilitates the use of Simulink, which is a block-diagram programming environment to simulate dynamical systems. Next, WinLib provides Simulink compatible real-time drivers to access the MultiQ-3 DACB. Once a block-diagram is built in Simulink, its architecture is automatically converted to real-time C-code using RTW following which the C-code is compiled using Visual C++ to generate an executable controller file. This real-time controller file is implemented using WinCon,[14] a client/server software environment that communicates with the MultiQ-3 DACB. In particular, WinCon client executes the real-time process. Furthermore, WinCon server communicates with WinCon client to acquire signal information, associated with the block elements in the Simulink block diagram for display into virtual instruments such as scopes, plots, digital meters, and thermometers. Finally, WinCon allows updates of real-time controller parameters during program execution by *i)* using Matlab script, *ii)* user interaction through the Simulink diagram, and *iii)* using a WinCon server GUI called Control Panel.

C. Control Design/Analysis/Implementation

Matlab's controls toolbox aids students to design/analyze a variety of controllers, e.g., proportional-integral-derivative controllers, linear quadratic regulators, observer based controllers, etc. In addition, Matlab's scripting language is used for "on-the-fly" controller parameter tuning. Real-time controller implementation and validation using Simulink, RTW, WinLib block-diagram programming architecture, and a rapid code generation process, enhance students' laboratory experience by allowing them to focus their efforts on control design rather than on programming and debugging in lower level languages such as C.

D. Web-Based Online-Experiments

The objective of the MPCRL web site[1] is to facilitate student learning by providing an information and navigation center as shown in Figure 3. As an information center, the MPCRL web site provides students information/installation tips on the software needed to run online-experiments. The MPCRL web site also provides important notices, technical documentation on modeling and control of experiments, and recent updates. As a navigation center, the MPCRL web site provides students useful links to other web sites, contact information of professors and teaching assistants, and a navigation bar to move between web pages of each online-experiment. As an educational enhancement, the MPCRL web site provides remote access to six experimental test-beds (as of this writing, see Figure 4 and Table 3) through the Internet by means of a secondary software environment. Figure 2 shows the hierarchy of the secondary software environment that utilizes WebLab,[15] the MPCRL web site, and Java applet driven control panels.

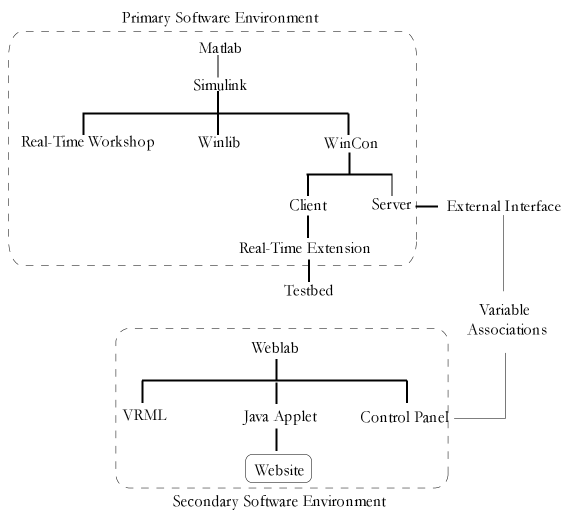

Figure 2: Software environment

Figure 3: The MPCRL homepage

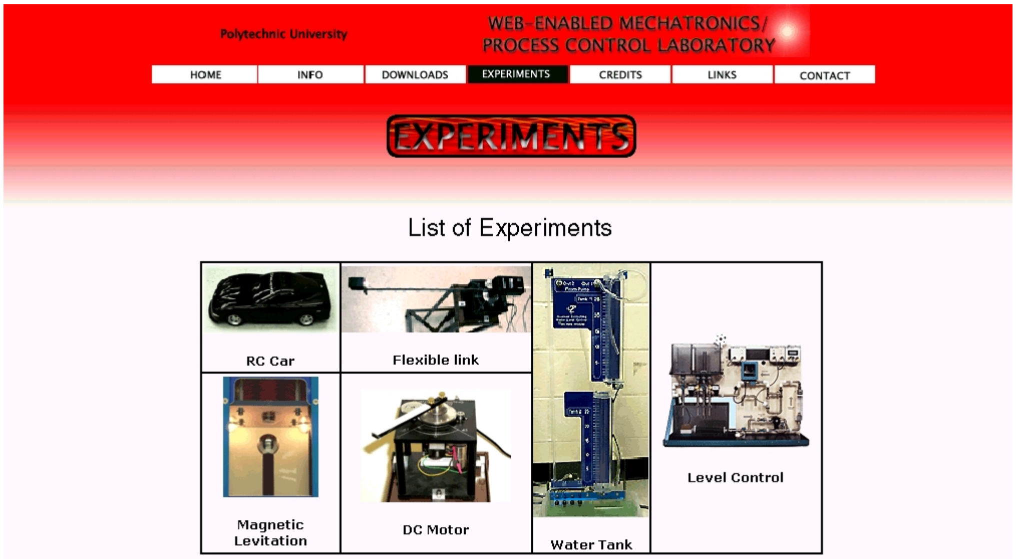

Figure 4: List of online-experiments

Table 3: List of online-experiments currently available on the MPCRL web site

Multidisciplinary Test-Bed	Description
DC Motor	Students learn to explore set-point angular position control of a link driven by a DC motor
Flexible Link	Students learn to control vibration of a rotary flexible link using linear quadratic control
Level Control	Students learn to apply control design techniques on a fluid process system to control the liquid level in a holding tank
Magnetic Levitation	Students learn to control the position of a steel ball using an electromagnet
Remote Control Car	Students learn to apply reverse engineering to drive a radio controlled toy car
Water Tank	Students learn to apply control design techniques on a two-tank system to control the liquid level in the lower tank

WebLab,[15,16] developed by Quanser Consulting Inc.,[17] is a GUI-based application that is used to create control panels (or virtual instrument panels), by placing control and display components (e.g., knobs, sliders, edit boxes, plots, 3D animations, etc.) on a blank canvas. The WebLab control panels are interfaced with the WinCon client/server environment to control and monitor experimental test-beds. Specifically, using WinCon server's external interface GUI, variables representing control and display components on the WebLab control panel are associated with the control and display blocks in the Simulink controller diagram. Next, the WebLab GUI is used to generate Java applets that are integrated within a web page to access the WebLab control panel remotely.

In a WebLab control panel, control knobs, edit boxes, and switches, defined as *active elements*, are associated to reference signals, control gains, and control switches in the Simulink controller diagram. The active elements can allow remote control of the experiment by sending data packets to a WinCon client/server. To observe basic motion of an experiment, 3D virtual worlds are created using a Virtual Reality Modeling Language (VRML) editor supplied with WebLab. Basic 3D shapes like cubes, spheres, cylinders, cones, etc. are assembled together with motion constraint joints (connections that restrict translational/rotational motion between shapes) to approximate the shape and movement of an experiment. Next, 3D animations are inserted into control panels similar to active interfaces. In addition, data plots in a control panel allow monitoring of physical variables. These visual displays, inclusive of a live video stream in a side window, are defined as *passive elements*. Students use the remote control panels following the hands-on experience with the experimental test-beds. The web-enabled control panels can be accessed using any Internet browser after downloading and installing two Java plugins as delineated in a set of instructions on the MPCRL web page.[1]

When a remote user starts an online-experiment through a web-enabled control panel, the real-time control algorithm is implemented on the web-enabled client PC. Remote users have the ability to *i)* manipulate the active elements to change controller parameters and reference signal parameters and *ii)* monitor the online-experiment's response via the passive elements. Whereas a remote user's input to an active element is communicated to the web-enabled client PC instantaneously (subject to prevailing network delay), the state of the online-experiment is communicated by the web-enabled client PC to the remote user's control panel using batch processing. Using control switches in a web-enabled control panel, remote users have the ability to select a control architecture from among the given preprogrammed control architectures. For example, in the flexible link experiment control panel shown in Figure 5, a tip control on/off switch is provided to demonstrate the degrading effect of neglecting the flexible link dynamics in the control design process. This allows students to explore and compare the effectiveness of competing control designs on an experimental test-bed.

E. Web Enhancements

Recent enhancements to the MPCRL web site include prerecorded movies of experimental test-beds, a Java chat program, and a supplementary virtual tutorial, which demonstrates various software elements used for: real-time data acquisition and control, web-enabled control panels, animation models, and live streaming video. The prerecorded movies of experimental test-beds introduce and demonstrate feedback control *in action* to uninitiated users and motivate them to explore further. The MPCRL web site provides a chat feature so that remote users connecting to the web site can communicate and interact with each other through text-based messages. This environment promotes students to vocalize their ideas freely and anonymously, without fear of being judged. Furthermore, teaching assistants and/or professors logging into the chat environment can interact with multiple remote users to help with important concepts pertaining to the online-experiments. As discussed below, even though only one user can manipulate the active elements of a web-enabled control panel at a given time, all the users connected to the experiment's web page can design controllers collaboratively, which can be implemented by the user having command of the control panel. This facilitates team-learning among the users. Finally, a supplementary virtual tutorial teaches the interested remote users the procedure of developing online-experiments. Using this feature, users learn to make their experimental test-beds web-enabled so that they can access their laboratory hardware remotely.

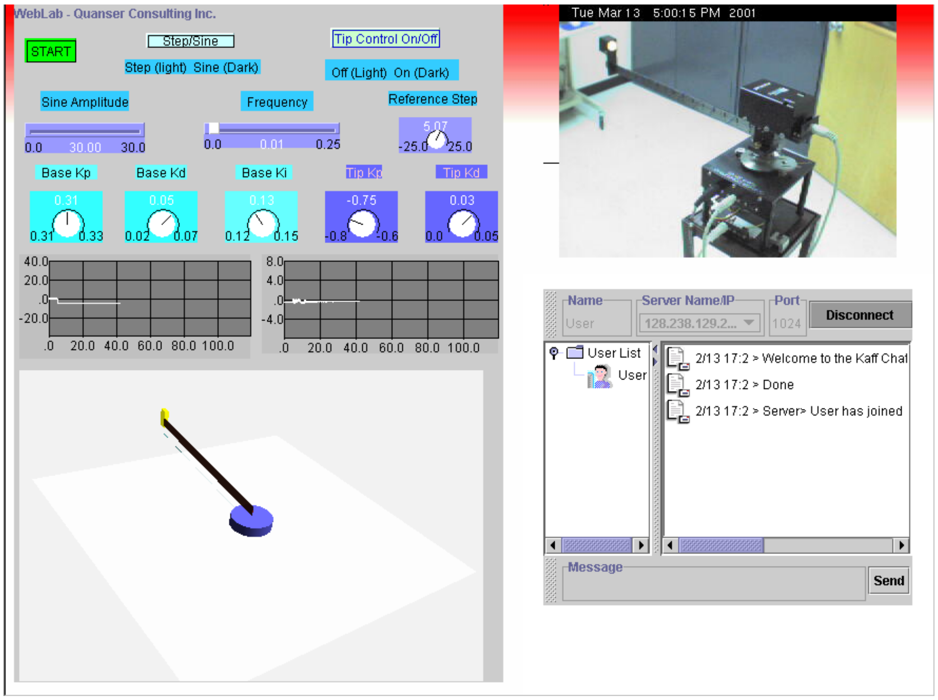

Figure 5: Web-enabled control panel for flexible link experiment

F. Network Configuration

A suitable network configuration was used to facilitate a bandwidth intensive web site. Figure 6 shows the local network configuration used for the MPCRL web site. A web server PC hosts all web pages for this web site and connects directly with the Polytechnic University Internet domain (PUID). When a remote user opens a control panel window on a specific online-experiment web page, the web server PC directs the remote user’s web browser to the appropriate web-enabled client PC that connects with the online-experiment. Using this redirection, the computational effort in running the online-experiment or providing the video stream is placed duly on an individual web-enabled client PC. On the remote user’s PC, Netscape Communicator version 4.6 is the suggested web browser for viewing the MPCRL web site. Recent versions of Internet Explorer are also supported (providing that the appropriate Java plugins are installed).

Students working on an online-experiment may encounter data transfer delays in receiving data packets or in viewing the live video stream. These delays depend on the prevailing network congestion, central processing unit (CPU) speed on the remote user’s PC, and type of Internet access connection. Although, not much can be done to alleviate network congestion delays, remote users should try connecting to the MPCRL web site during off peak hours. In addition, a set of minimum and recommended PC configurations is given in Table 4. Remote users running an online-experiment with direct access to PUID should experience minimal data transfer delays. However, remote users connecting from outside the PUID using 56 kilobits per second modems (or slower) will notice significant delays primarily due to the live streaming video.

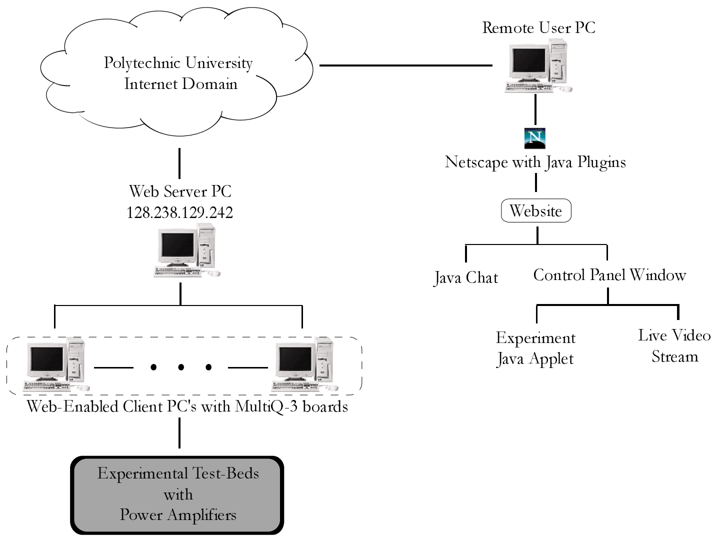

Figure 6: Network configuration

Table 4: List of minimum and recommended PC configurations for using online-experiments

	Minimum Configuration	Recommended Configuration
CPU speed	200 Mhz Pentium II processor	500 Mhz Pentium III processor
System Memory	64 megabytes	64 megabytes
Video Memory	4 megabytes	16 megabytes
Network Hardware	56k modem	Network Interface Card
Resolution	800 by 640 pixels with 16-bit colors	1024 by 760 pixels with 32-bit colors

Online-Experiment Safeguards

Providing online-experiments as a learning enhancement tool is beneficial for students; however, safeguarding online-experiments is a major concern. Without supervision by professors and teaching assistants, online-experiments are susceptible to damage in the absence of safety features. Figure 7 shows the types of safety protocols used in a typical online-experiment in the MPCRL. These precautions, divided into hardware and software safety protocols, can be applied on physical experimental test-beds and within control algorithms or web-enabled control panels, respectively.

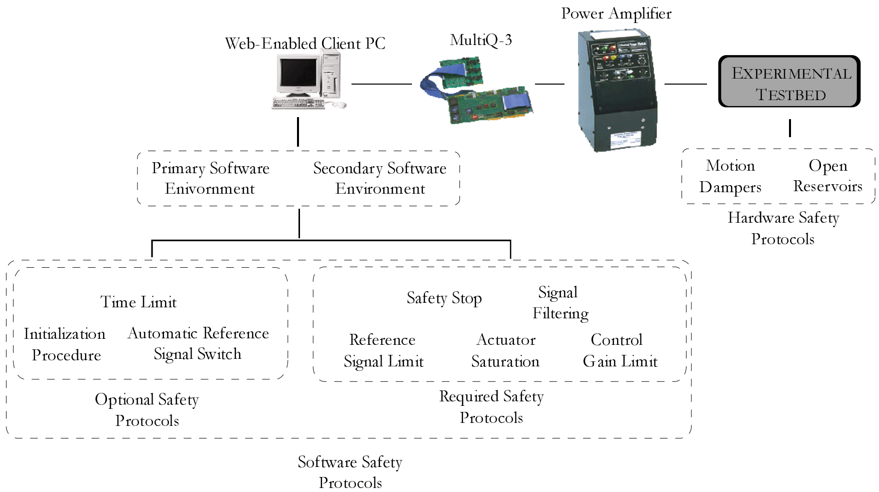

Figure 7: Safety Protocols

A. Hardware safety protocols

Hardware safety protocols consist of devices that limit the range of "motion" for experimental test-beds. Outpour of effluent agents or fluid, which can cause damage by overflow to surrounding equipment, is a type of motion for chemical-thermo-fluid systems. In order to restrain outflow of damaging effluents/fluids, drainage pipes are used to remove excess liquid from a fixed volume reservoirs. On the other hand, movement of linkages, which can cause damage to internal components and surrounding equipment through impact loading, is a type of motion for kinematic systems. In order to constrain the motion of linkages, limit switches, physical passive damping elements (e.g., rubber stops), etc., are placed along "sensitive" areas of experiments.

B. Software Safety Protocols

Software safety protocols provide greater flexibility to safeguard online-experiments. In Figure 7, we assign two types of software protocols, namely required and optional. Required safety protocols consist of mandatory software components, implemented within the primary and secondary software environments, that prevent hardware limits (i.e. violations of output voltage limitations, spillover of high frequency actuation signals, etc.). These components can be in the form of saturation blocks, signal filters, parameter limits, etc. Of these, saturation blocks and signal filters are supplied within the Simulink block-diagram while parameter limits are used in control panels. Optional safety protocols consist of voluntary software components, implemented within the primary and secondary software environments that help display the purpose of an experiment and to promote fairness among users connecting to an experiment.

C. Connectivity Queues

Usually, control laboratories are limited to a fixed number of experiments that are less than the total number of students enrolled. Similarly, the MPCRL can provide only a fixed number of online-experiments at a given time. In response, control panels created through WebLab provide connectivity queues that allow only one person to manipulate its active elements at a given time and safeguard against users attempting monopolistic use of an experimental test-bed.

If no one is connected to an online-experiment control panel, the first remote user to connect to that experiment becomes the *primary user*. The primary user can manipulate active elements of a web-enabled control panel. When an experiment is in use, other remote users connecting to the same online-experiment control panel are called *secondary users*. The secondary users cannot manipulate active elements of a web-enabled control panel. However, both the primary user and the secondary users can monitor the online-experiment via the passive elements. That is, they can view the real-time experiment plots, 3D animations, and live streaming video frames. When a primary user disconnects from the online-experiment control panel, a secondary user can become the primary user by reloading the control panel. At the current writing, the primary users are permitted to keep control of an online-experiment for 10 minutes at a stretch after which they are automatically transferred to the MPCRL homepage without any warning. The purpose of these guidelines promotes fairness in the use of online-experiment and prevents monopolistic overuse of laboratory resources.

Evaluations and Comments

To assess the efficacy of the MPCRL, comprehensive evaluations are conducted using a formal mid-term and end-of-the-term questionnaire. The students provide anonymous responses to the questionnaires as well as individual comments and feedback on their laboratory experience by means of open-ended questions. The questionnaire is designed following the guidelines of the ABET 2000 criteria on outcome-based-assessment. These surveys provide quantitative and qualitative data that aid in the improvement of laboratory pedagogy. For example, students are surveyed on the skills acquired in interfacing sensors and actuators with DAC boards; step response-based system identification; formulating, analyzing, designing, and implementing classical controllers; analyzing and interpreting data; etc. Furthermore, we asked students "What did you find most valuable about this course?" Some sample student remarks were:

- The hands-on approach in the control laboratory helps "connect theoretical knowledge" to real-world applications.
- The laboratory experience exposes us to an experimental methodology that provides "hands on" interaction with hardware and teaches us to "simulate and control physical processes."
- Performing the laboratory experiments helps learn "troubleshooting" techniques in real-world mechatronic systems and shows "controls in action."

Some constructive criticisms provided by students included:

- Laboratory experiments should be conducted in parallel with the Automated Control course.
- The laboratory experiment should not restrict students to design fixed control architecture.

Conclusion

In this paper, we have discussed the development of a multidisciplinary control laboratory paradigm that promotes competencies in project-based, cooperative, active, and peer learning. Using standardized data acquisition hardware and experimental methodology, students interact with multidisciplinary test-beds, validate abstract concepts, and learn integration of system components. A unified software environment provides students modern programming tools to perform rapid control design, analysis, implementation, and validation on a broad array of multidisciplinary test-beds. In addition, the MPCRL web site has been created, which includes online-experiments and web enhancements, as an innovative educational tool that provides students unlimited access to laboratory hardware and promotes student-centric learning. To address the safety issues in providing online-experiments, sets of software and hardware safety protocols have been developed. The MPCRL has been used within the past year, for undergraduate, graduate, and high school student education and research. In particular, 23 undergraduate and 22 graduate students from Polytechnic University, 7 graduate students from other universities, and 7 high school students have enrolled/participated in the MPCRL educational/outreach activities.

Acknowledgements

The financial support for the laboratory development activities described in this paper has been provided in part by the NSF—DUE CCLI—Adaptation and Implementation Grant 9950782 and the NASA/New York Space Grant Consortium grant 32310-5891. The authors wish to acknowledge the additional financial support provided by the Office of the Dean of Engineering (Prof. William McShane) and the Mechanical, Aerospace, and Manufacturing Engineering Department (Profs. Sunil Kumar and Said Nourbakhsh), Polytechnic University, Brooklyn, NY. The authors are thankful to Prof. Marcio S. de Queiroz and students Haizhou Pan, O’neil White, and Qiguo Yan for their assistance throughout the development of the MPCRL. The authors thankfully acknowledge the support of suppliers of DSP and PC boards, equipment, and instrumentation, namely ECP, dSPACE Inc., Feedback Control, and Quanser Consulting Inc. Finally, the authors gratefully acknowledge the support of Quanser Consulting Inc. in providing the WebLab technology.

References

1. URL: http://128.238.129.242; Mechatronics/Process Control Remote Laboratory.
2. URL: http://mechanical.poly.edu/faculty/vkapila/ControlLab.htm; Real-Time Mechatronics Process Control Laboratory.

3. V. Kapila, M. S. de Queiroz, and A. Tzes, “A Multidisciplinary Undergraduate Real-Time Experimental Control Laboratory,” *Proc. Amer. Contr. Conf*., Chicago, IL, June 2000, 3980—3984.
4. URL: http://mechanical.poly.edu/faculty/vkapila/YES.htm; YES Scholar Program.
5. URL: http://mechanical.poly.edu/faculty/vkapila/workshop.htm; Real-Time Experimental Control Workshop.
6. *MultiQ-3 Programming Manual*, Quanser Consulting Inc.
7. V. Kapila, *Laboratory Experiment Manual for ME 325: Control Laboratory*, Polytechnic University, Brooklyn, NY, 1999.
8. URL: http://mechanical.poly.edu/faculty/vkapila/quanser.htm; Equipment Manuals.
9. URL: http://www.fbk.com/full/products/prodfrmpdf.htm; Feedback Instruments Limited: Home Page.
10. URL: http://www.vci.com/products/vci_products/rtx/rtx_index.html; VenturCom-RTX Products.
11. URL: http://www.apache.org/; The Apache Software Foundation: Home Page.
12. URL: http://www.axis.com/; Axis Communications: Home Page.
13. URL: http://www.mathworks.com/; The Mathworks: Home Page.
14. *WinCon User’s Manual*, Quanser Consulting Inc.
15. J. Apkarian and A. Dawes, “Interactive Control Education with Virtual Presence on the Web,” *Proc. Amer. Contr. Conf*., Chicago, IL, June 2000, 3985—3990.
16. *WebLab User’s Manual*, Quanser Consulting Inc.
17. URL: http://www.quanser.com/; Quanser Consulting Inc.: Home Page.

HONG WONG was born in Hong Kong, China. In June of 2000, he received a B.S. degree in Mechanical Engineering from Polytechnic University, Brooklyn, NY. He is a member of Pi Tau Sigma and Tau Beta Pi. He worked for Air Force Research Laboratories, Air Vehicles Directorate, in Dayton, Ohio, during summer of 2000 and is currently a graduate student at Polytechnic University. His research interests include nonlinear control of electromechanical systems and distributed parameter systems.

VIKRAM KAPILA is an Assistant Professor of Mechanical Engineering at Polytechnic University, Brooklyn, NY. He has held visiting positions with the Air Vehicles Directorate of Air Force Research Laboratories in Dayton, OH. His research interests are in linear/nonlinear control with applications to robust control, saturation control, closed-loop input shaping, distributed spacecraft formation flying, spacecraft attitude control, and mechatronics. He has organized several sessions at international conferences and is active in the development of a strong experimental control program at Polytechnic.

ANTHONY TZES is active in the area of adaptive, fuzzy logic, and neural control, with applications to robotics and transportation systems. He is a guest scientist at BNL, a member of IEEE’s conference editorial board, and an associate editor of IEEE’s Control Systems Magazine. He was responsible for the control program at Polytechnic University (1990-96) and the development of the undergraduate Instrumentation and Control laboratory and the capstone design sequence in mechatronics.