Lessons Learned from 10 Years Experience with Remote Laboratories

James TREVELYAN
School of Mechanical Engineering, The University of Western Australia, 35 Stirling Highway, Crawley 6009, Perth, Western Australia, James.Trevelyan@uwa.edu.au, http://www.mech.uwa.edu.au/jpt/

KEYWORDS: *remote access laboratories, internet, student learning, transfer effectiveness*

ABSTRACT: *Remote access laboratories using internet communications have been available in prototype form since the mid 1990's (Goldberg 1999, Taylor and Trevelyan 1995, Henry 2003). We started our on-line laboratory facilities at the University of Western Australia in October 1994 with the commissioning of Australia's Telerobot on the Web.*

This paper reviews all the principal lessons we learned since 1994 and briefly describes Telelabs, a cost-effective framework to provide an extendable series of on-line labs that can be sustained from normal operating budgets. The Telelabs framework was commissioned in 2002 and at about 5% of the reported cost of two other comparable frameworks (I-Lab at MIT and PEARL at Open University). While maintenance costs are still a little too high for long term operation, they are steadily being reduced.

The system currently supports five different remote labs, including the original Telerobot that was re-engineered within the Telelabs framework in 2003:

- *Instrumented domestic electric iron: thermodynamics, mechatronics and control*
- *Sand weighing machine: bulk materials handling and PLC control*
- *Position control: classical control systems*
- *Torsional vibration: dynamics and vibration control*
- *Telerobot: robot kinematics and control*

From 2002 onwards we have focused on designing effective learning activities for students using on-line labs and evaluating learning effectiveness.

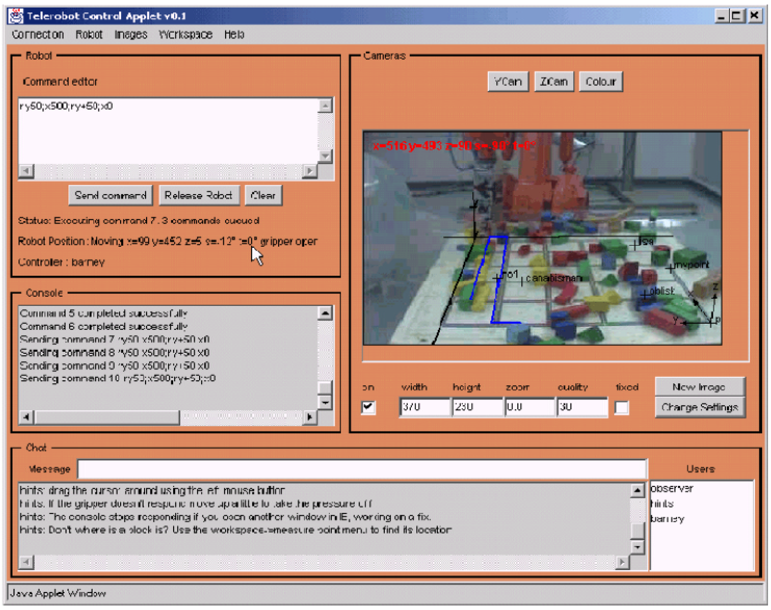
The paper reports the results of recent trials with remote on-line laboratories as part of weekly tutorial exercises for mechatronics engineering undergraduates. These trials reveal that students using remote access laboratories operated equipment for much more time than in conventional laboratory classes, and learning outcomes seem to be significantly improved as a result. This allows students the time they need to explore the differences between theory and real equipment behaviour for themselves. With the time constraints of normal lab classes, students mostly tend to follow a pre-determined sequence of measurements, calculations and written reporting without necessarily understanding what they are doing. The style of interaction with equipment using remote access laboratories is more representative of typical industrial scenarios.

1 TELEROBOT (1994 – 2004)

The Telerobot (Taylor and Trevelyan 1995, Dalton 2001) was among the first web-controlled devices publicly available on the internet. My students overcame a challenging technical problem: developing a means of providing public access to allow anyone with a web browser to operate a powerful and complex robot remotely with reliable long-term operation and effective protection of the robot from accidental damage. This included shared operation to allow a number of users to collaborate effectively, sharing control of the robot between them. This research yielded many valuable lessons for routine remote operation of laboratory equipment.

We used the Common Gateway Interface method (CGI) for early implementations of the telerobot. However this relies on the remote user to initiate all communications and is quite unsuitable for processes that generate continuously changing results. Dalton (2001) implemented a Java applet client that provided asynchronous communication between the robot and the remote user. Using measurement code built into the robot control applet he conducted a series of communication tests using different Internet communication protocols. Given the large number of users interacting with the robot (approximately 2000 per week) he obtained several useful and statistically significant results. He found that TCP/IP socket connections open by remote clients offered the most reliable connections. 90% of users could use unregistered ports for this. While 'channelling' data through port 80 (used for web page data in HTTP) would permit most of the remaining 10% to access the robot, these people tended to be located behind company firewalls and would not be typical of university student users. Although UDP/IP is faster in theory for transmitted data this was seldom the case in practice. UDP/IP connections were blocked in many cases by firewall restrictions. The round-trip time, the time taken for a message from a remote client to be received by a server and be retransmitted to the client, a significant parameter for online remote control applications, was about one second and was independent of message length for messages less than 1000 bytes long. Lower round-trip times were only feasible within national or local high-speed area networks. Finally, although there were substantial improvements in Internet traffic capacity through the life of the project there was little significant improvement in bandwidth for dynamically refreshed data.

Dalton (2001) also considered different configurations for servers and clients. He showed how better reliability could be achieved using *message-oriented middleware* (MOM) server configuration to reduce the number of connections that had to be sustained, and allow recovery from a temporary connection loss. He also pointed out that the server can support almost all communications functions without having to have any knowledge about the technical details of the equipment being operated remotely. Hardware specific software elements are more easily implemented as "hardware" and "remote user" clients that connect to the server.



ry50;x500;ry+50;x0

Status: Executing command 7. 3 commands queued

Robot Position: Moving x=99 y=452 z=5 s=-12° t=0° gripper open

Controller : barney

```
Command 5 completed successfully
Command 6 completed successfully
Sending command 7 ry50 x500;ry+50 x0
Sending command 8 ry50 x500;ry+50 x0
Sending command 9 ry50 x500;ry+50 x0
Sending command 10 ry50;x500;ry+50;x0
```

☑ width 370 height 230 zoom 0.0 quality 30 fixed □

```
hints: drag the cursor around using the left mouse button
hints: If the gripper doesn't respond move up a little to take the pressure off
hints: The console stops responding if you open another window in IE, working on a fix.
hints: Don't where is a block is? Use the workspace->measure point menu to find its location
```

observer
hints
barney

Figure 1 – Java applet interface to Telerobot (Dalton 2001)

When Dalton started in 1996 Java seemed the ideal environment for developing software for distributed applications on the Web. However, he discovered that Java applets running in web browsers could not provide a reliable platform for complex remote clients, unlike the MOM server which was built as a standalone Java application and operated reliably. There were two problems. First, different combinations of browser, operating system, graphics hardware and processor would result in different (and sometimes incorrect) appearance of the applet. Second, there were many problems with the stability of the Java engine in different browsers. In the end it was only possible to obtain satisfactory operation if users ran the applet in a specified version of Internet Explorer. There was a significant download penalty for users: the applet had to be downloaded every time a user wanted to access to the telerobot. Finally the complexity of the code and dependency on specific hardware interfaces led to unsustainable maintenance costs.

2 REMOTE LABORATORIES FRAMEWORK (2000 – 2004)

The Telerobot experience led to our Telelabs framework for remote access laboratories.

In 1999 the university provided a development grant for a remote laboratories project that would also provide a challenging learning environment for Mechatronics engineering students. With the knowledge that Java could not offer a reliable development platform we searched for industrial alternatives. We were looking for facilities to construct distributed Internet applications with convenient code and a wide range of hardware interface options. We wanted to avoid an upgrade obsolescence problem that often arises when a particular piece of hardware needs to be replaced. It is not always possible to operate the replacement hardware in an old operating system and it is often difficult to avoid re-engineering the software.

In the long term we were also looking for ongoing commercial support at reasonable cost and, with potentially thousands of student users, we wanted to avoid a solution that required a software license for every user.

We investigated the products offered by several companies specialising in industrial control systems and instrumentation such as Honeywell, Citect, MathWorks, National Instruments Motherwell, Leeds Northrup and others. We selected LabVIEW because it has excellent Internet connectivity, supports a wide range of industrial hardware, has multiplatform support, license cost independent of the number of users (for compiled executables) and is well-known in universities.

LabVIEW and MATLAB were beginning to provide web enabled applications for remote access when we started the project. However, even now four years later with mature technology, neither solution offers administrative support necessary for effective remote access laboratories.

Between 2000 and 2003, therefore, we invested considerable development funds to develop Telelabs, a new extendable server and remote labs framework based on commercial software components (based on LabVIEW) that would lead to sustainable on-going maintenance and development costs. We commissioned Icon Technologies, a local centre of expertise in LabVIEW, to engineer our remote access laboratories server and client technologies incorporating

- routine monitoring of student actions
- access controls appropriate for a university operating environment,
- queuing for students waiting to access hardware,
- task-dependent user interfaces for remote users,
- collaborative shared control for group learning,
- built-in archiving of experiment data,
- recovery from lost connections, and,
- timetable reservations.

We designed our framework to support the four main kinds of remote lab experiences which have been reported in the literature:

1. Queued batch: the user sets parameters and transmits a command to begin the experiment. There is no user interaction during the experiment. Either the experiment happens so quickly that interaction is impractical, or interaction is undesirable, or is simply not provided for in the software. (see for example Henry 2003, del Alamo 2003)
2. Real time interactive: the user can change parameters and observe results in real time. The electric iron experiment described later in the paper is a representative example. There are three limiting factors:
    - The network round trip time: the time taken for a command to be transmitted to the equipment and the initial response to arrive at the user's computer to be displayed – this is typically between 0.1 and 1 sec.
    - The network bandwidth may restrict the type of feedback available to the user, and the rate at which it can be displayed.
    - Streaming video or sound will be delayed by several seconds due to buffering at the user's computer.
3. Real-time measurements without control (typically with archives of previously recorded data), for example the MIT Flagpole (Amaratunga and Sudarshan 2002), and weather stations. In this kind of experiment there is no need for the user to set controls except, perhaps, to select the measurements and data rate. Transmission time delays (up to a few seconds) are unimportant and there is the possibility of synchronizing data with a video stream.

4. Programming experiment. In this kind of experiment, the user is expected to develop code for a programmable device. While editing and reviewing the code there is no need to access the equipment. When ready, the code needs to be executed, possibly with some kind of user interaction. The execution phase can be type 1, 2 or 3 above (Type 2 example at end of paper).

We designed the Telelabs framework so that the skills and experience of our undergraduate mechatronics engineering students would be sufficient for them to carry out most of the on-going system development work as part of their engineering project work. Icon Technologies continue to supply ongoing maintenance support for the server software.

Several other universities have commissioned similar remote laboratories framework projects in recent years. They include the Swiss Federal Institute of Technology (Salzmann *et al* 1999), Massachusetts Institute of Technology (de Alamo *et al* 2002, 2003), the British Open University (Schäfer *et al* 2002) and at least two consortia of German universities and technical colleges (see Böhne *et al* 2002). The MIT and Open University projects have multi-year budgets of approximately US$3,000,000 each. However, as a result of careful design, extensive prior research, and perhaps the selection of LabVIEW as a software development environment we have managed limit our expenditure to around US$130,000, of which approximately 40% was spent on server software development. The MIT project is being engineered in Java and .NET and the Open University project is engineered in Java.

3 CURRENT REMOTE EXPERIMENTS

The current suite of remote labs offered (at appropriate times) includes the following list (see also Trevelyan 2002). Students are working on several more experiments that we will introduce in future.

a) Instrumented domestic electric iron: thermodynamics, mechatronics and control (Types 2 and 4)
b) Sand weighing machine: bulk materials handling and PLC control (Types 2 and 4)
c) Position control: classical control systems (Type 1)
d) Torsional vibration: dynamics and vibration control (Type 2)
e) Telerobot: robot kinematics and control (Type 1, will include 4)

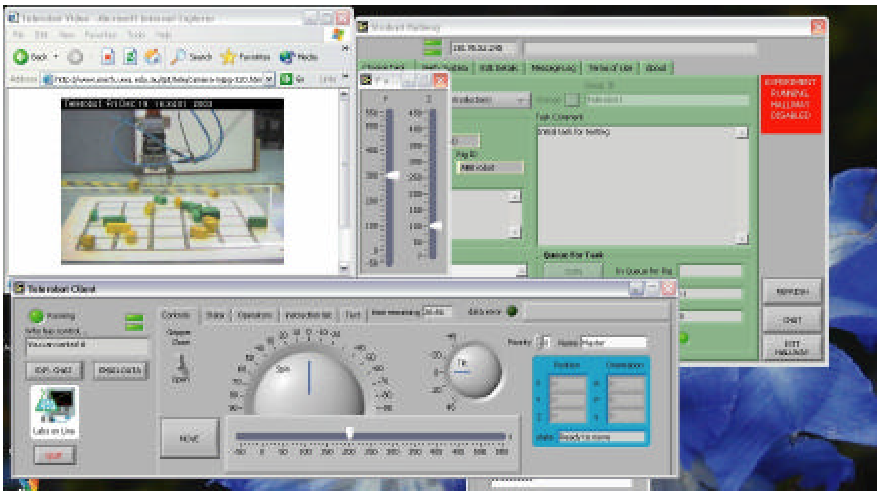


Figure 2 – Telerobot interface in Telelabs framework (Babbage 2003).

4 EFFECTIVE LEARNING TECHNIQUES: 2003 TRIAL

Much more has been written on the technologies behind remote access laboratories than on effective ways to make use of them.

Remote labs cannot serve as a direct replacement for conventional lab classes because manual intervention or manipulation is not possible and direct sensations have to replaced with various forms of information display. On the other hand remote labs offer students more opportunities to use equipment in closer synchronisation with lecture and tutorial presentations because the normal timetabling constraints do not apply. This means that remote lab learning activities need to be designed carefully to use their advantages to full effect. Given that the ultimate object of laboratory-based learning is to help students acquire the skills needed to conduct engineering investigations on real equipment and engineering systems, the concept of transfer effectiveness is relevant (Roscoe and Williges 1980). This concept enables the utility of different student practice tasks: theoretical paper-based studies, simulation, remote access laboratories and conventional laboratory classes to be seen in terms of the reduction in practice time needed for skill development. Each learning mode can contribute some understanding and practice. In each case, from research results in aviation human factors one can expect that the degree of fidelity and realism reproduced in the practice tasks greatly influences the transfer effectiveness. It is not easy to measure transfer effectiveness, but it is a useful framework in which to think of the different contributions to learning that each practice task can make.

A typical conventional laboratory class for engineering students lasts three hours. Realistically we do not assume students have any significant understanding of the equipment or the laboratory exercise at the start. Between two and six students, working as a group, will follow a pre-planned series of observations and measurements and may also write a brief report before they finish. Observations in our own laboratory classes (Davies 2002) suggested that the average time for which any individual student operates the equipment is approximately 20 minutes and as many as half the students do not operate the equipment at all, preferring instead to watch other students.

Engineers working in an industrial context are expected to interact with equipment and solve problems in quite a different style. Typically they have several weeks in which to familiarise themselves with the equipment and the problem to be solved. They devote effort to other activities and priorities at the same time and they do not have a prescribed experimental plan to follow. They are expected to devise their own plan of approach. They will typically take observations and recordings over, perhaps, several weeks and then experiment with different solutions. The time available for experiments is probably tightly restricted by the need to minimize any loss of 'productive' operation time. Eventually, an engineer may install and commission some improvements to the equipment and conduct some further tests to confirm that anticipated improvements have been realized.

A remote access laboratory offers the opportunity for students to interact with equipment in a way that is much closer to this kind of typical industrial situations. Students are free to use the equipment when their other priorities permit (subject to demand from other students), and can take their own time to formulate useful experimental approaches. Occasional visual inspection of the actual equipment may be necessary.

We have pursued two types of learning activities using remote labs. First, remote labs are offered as an optional supplement to normal lab classes: students can take their own observations and explore equipment behaviour that can be observed remotely. The second type consists of a series of short (typically 15 minute) tasks performed as part of weekly assigned exercises, with a gradual increase in task complexity and difficulty. This seemed to be more successful from 2003 evaluations.

Our remote online laboratory system (Trevelyan 2002, 2003) still enables students to perform the laboratory exercises in the laboratory alongside the actual equipment, just as they would in a normal laboratory class though without direct supervision. We found that many of the students preferred to work remotely.

By using a remote online laboratory system we can subdivide the laboratory work into a series of short weekly exercises taking between 15 minutes and half an hour to complete. This way the online laboratory exercises can be integrated with weekly tutorial classes and lectures: each weekly exercises explores aspects of material covered in lectures in the same week.

Figure 3 illustrates the remote controlled equipment used for the 2003 trials: a domestic electric iron fitted with temperature sensors, a solid-state relay to switch the electric power on and off and a compressed air jet that can be turned on the cool the base plate of the iron when needed. 44 second year mechatronics engineering students were expected to learn about discrete (on/off) control equipment, thermal process dynamics, pulse width modulation to convert a discrete input into a pseudo-continuous regulation input, model and parameter identification for process simulation, and finally closed loop feedback control. A traditional three-hour laboratory session, even under ideal circumstances, would barely be sufficient for them to be exposed to these concepts, let alone understand them.

A domestic electric iron seems a very simple piece of equipment for mechatronics students to work with but it fulfils two essential requirements. First, it is safe to assume that all the students, male and female, are very familiar with what an electric iron does (even though the state of their clothes would suggest that they seldom if ever use one themselves). Our earlier experience with the Telerobot (Dalton

2001) revealed that many people never gained a complete understanding of the equipment and its layout in the laboratory even though they could access dynamically updated camera images and high-resolution archive digital photographs. Familiarity with the equipment at the other end of the remote experience seems to be a great help in understanding.

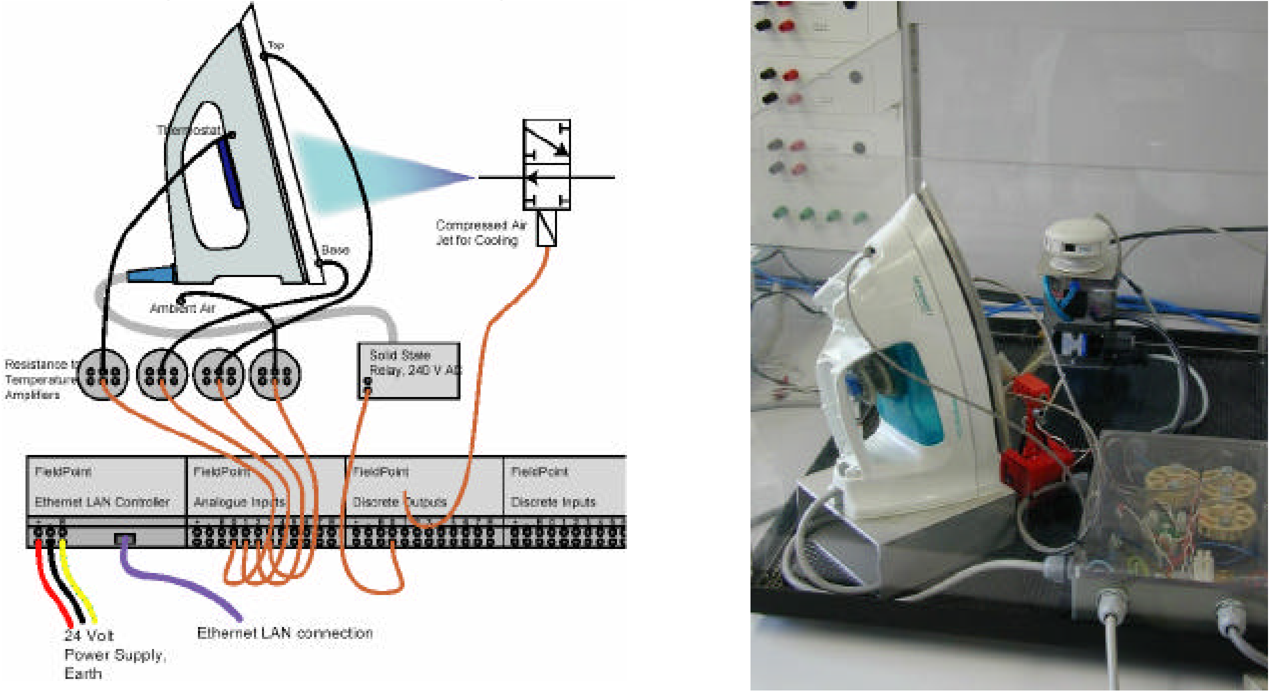
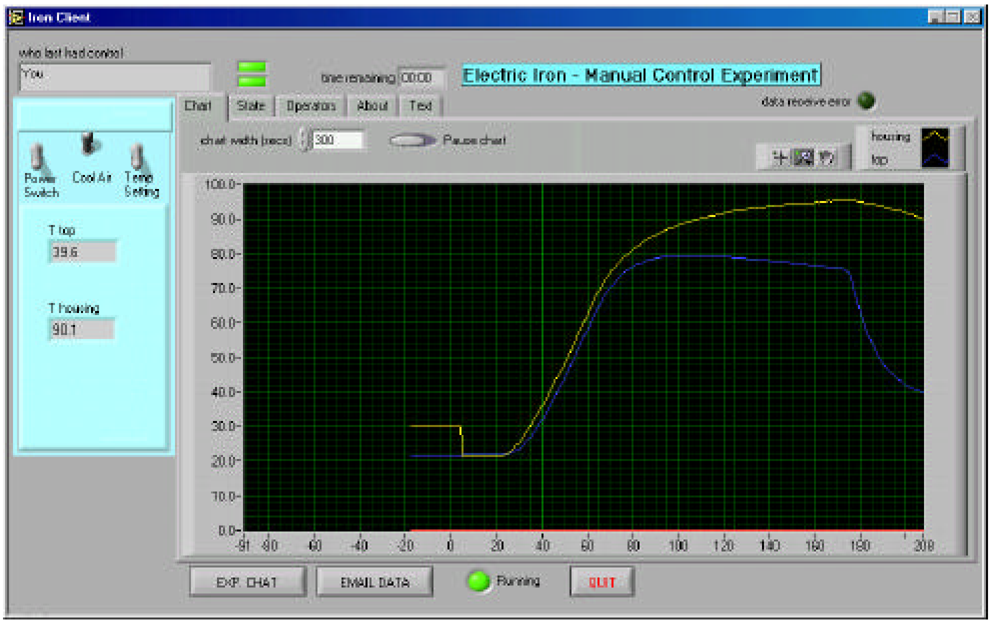

Top
Thermostat
Base
Compressed Air Jet for Cooling
Ambient Air
Resistance to Temperature Amplifiers
Solid State Relay, 240 V AC
FieldPoint Ethernet LAN Controller
FieldPoint Analogue Inputs
FieldPoint Discrete Outputs
FieldPoint Discrete Inputs
24 Volt Power Supply, Earth
Ethernet LAN connection

Figure 3a, 3b – Domestic electric iron used for remote laboratory exercises

The second essential requirement is that the equipment is simple, on the one hand, yet exhibits interesting subtleties of behaviour that only become evident when the students pay close attention to the temperature changes recorded by the sensors installed on the base plate of the iron. No visual feedback is necessary: there is no change in the visual appearance of the iron.



Iron Client

who last had control: You

time remaining 00:00

Electric Iron - Manual Control Experiment

chart width (secs) 300

T top 39.6

T housing 90.1

Figure 4 – Software client interface for students to control iron remotely.

The first exercise introduces students to the equipment and simple controls: a switch to turn the electric power on and off, another to turn the compressed air jet on or off and a third to set the thermostat knob to a high temperature or low-temperature position. (Figure 4) In the end, we decided not to use this third control at all: it was left in the high temperature position. In the first exercise students observe how the iron heats up when electric power is applied and how it cools, much more slowly, until the compressed air jet is turned on. If the compressed air jet is turned off after three or four seconds students observe and interesting phenomenon: the base plate of the iron gets hotter for the next 10 to 15 seconds before resuming a slow cooling trend. Most of the students are surprised by this: they realise that one cannot assume that the entire base plate is at the same temperature.

The second exercise introduces students to pulse width modulation control. Students operate the power switch and the compressed air switch manually using two or three different mark-space ratios and observe how the temperature rises more slowly. They also observe short-term fluctuations in temperature caused by the on-off switching.

In the third exercise pulse width modulation is automated (The task user interface has more extensive controls and displays to select pulse repetition period, mark-space ratio for air and power supply etc). Students observe the behaviour of the iron at different pulse repetition frequencies and mark-space ratios for both heating power and cooling air, chosen at random from a list of possible combinations supplied in a handout. Each student contributes his or her results to a shared forum building up a table of results covering all the combinations on the list. By this stage students have learned how pulse width modulation can be used to convert a simple discrete on-off control device into one that exhibits the possibility of infinitely variable control. This complements earlier experiences in their first-year classes when they were introduced to simple robot vehicles that used pulse width modulation to control the power going to the motors driving the wheels.

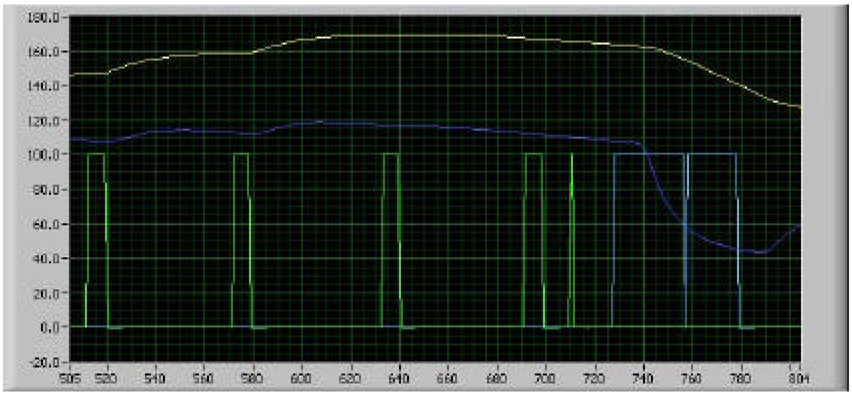


Figure 5: Typical student results from experiment 2: upper smooth traces – thermostat temperature, top of base plate temperature, lower discrete traces – power supply and air supply.

In the next week, the context changes to simulation: the students learn to create a simple first order differential equation model of the electric iron based on the data they obtained in the first week's exercise. They soon appreciate the deficiencies of this model and we provide them with a multi-element model illustrated in figure 6. We also provide them with code and a partially correct series of heat transfer coefficient and thermal capacity values for the elements. The students then use data from the experiments to improve this model through a trial and error process.

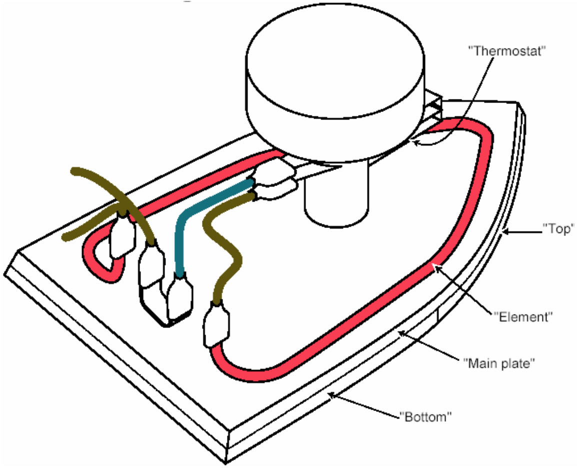


Figure 6 – Iron without plastic cover showing thermal "elements" used in multi-element simulation model.

After a two-week teaching break we supplied the students with code that enables them to test a prototype controller using the simulation model. They implemented their own controllers in LabVIEW within a skeleton template. A sample created by one of the students is shown in figure 7.

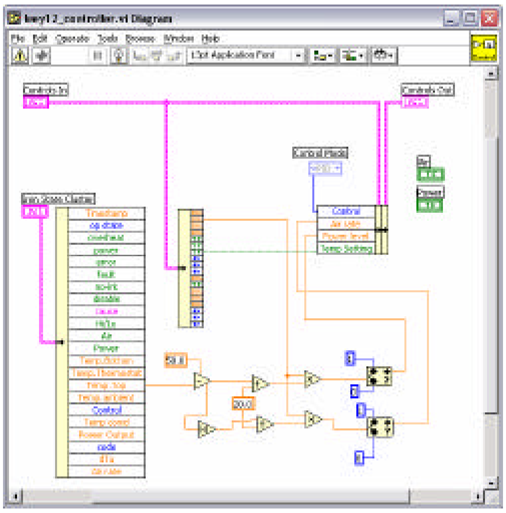


Figure 7 – LabVIEW diagram to implement typical student controller

Finally, the students return to the online equipment and test their controller using the actual hardware. Software limitations require the student controller code to be executed in the remote client of the online laboratory system. This means that there is a significant time delay (approximately 1.5 – 2 seconds). Also, many of the students did not bother to improve the deliberately incorrect simulation model by refining coefficients through trial and error. The simulated iron behaves differently but not to such an extent to make a simulation completely unrepresentative. Of much greater significance is the time delay introduced by the relatively long pulse repetition period students have programmed for the pulse width modulation controller (typically between two and 10 seconds). These time delays are not represented by the simulation model. Therefore, the behaviour of the student built controllers tends to be significantly different from what they expected.

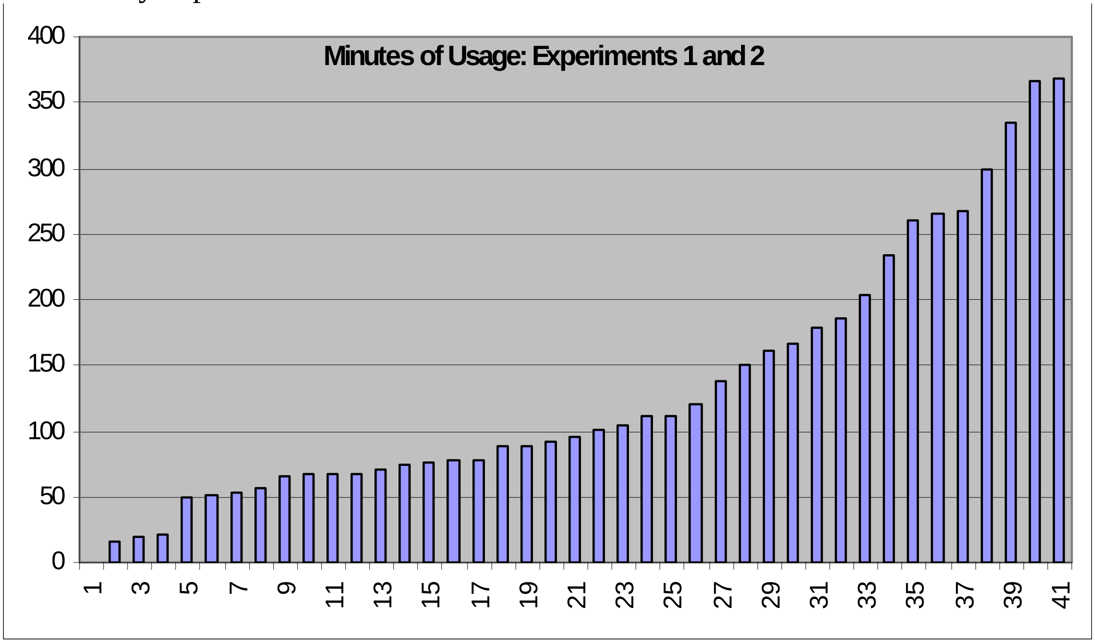


Figure 8 – This chart shows the actual operating time for exercises 1 and 2 recorded by the remote labs system for every student. Note that students who performed the exercises in the laboratory alongside the equipment also had to work the controls using the same remote laboratories interface so their operating time is also included in these statistics. Two students in the class did not attempt the exercises, one used the equipment for a minimal period of time and three students used the equipment for around 20 minutes total. The other students used the equipment for much more time than was necessary.

4.1 Unexpected Difficulties

Some difficult family circumstances necessitated my absence in Pakistan from the second week of this series of exercises until after the two week teaching break. A part-time tutor was able to help for the first two weeks of my absence but he too encountered circumstances that necessitated his absence. Fortunately, before my departure, the students had become accustomed to the use of a web based discussion forum for class communication, similar to WebCT. I was able to continue communicating with the students through this forum and individual e-mails. However, a further aspect of the remote laboratory system proved to be invaluable during this period. When the students perform their laboratory exercises the remote labs server sends recorded data to students by e-mail before they quit from the remote lab client. Any data transmitted by e-mail is also transmitted to the course controller by default. Therefore I was able to receive a full set of all the data recorded by the students and view this in Pakistan. With this data I could monitor the ways in which students were using the equipment and provide feedback to students having obvious difficulties within a few hours.

My absence made it impractical to enforce the suggested weekly deadlines. I permitted students to take their own time to complete the exercises with the predictable result that some students left their work until the last week of semester: the final deadline for the unit. A few software glitches caused difficulties for students who had to wait in the queue for their turn to operate the equipment: the students were given an extra few days to complete the exercises. Figure 8 shows that almost the entire class completed the first two exercises. 55% of the students completed the final online control exercise. The average time for which each student operated the equipment was 4.5 hours with a class participation of 96%. This needs to be compared with our earlier observations of conventional laboratory classes that suggest the average is around 20 minutes with only 50% class participation. Students received 9% of their total marks for this complete series of online and simulation exercises and typically, in other units, receive between 5% and 10% of the total unit marks for their laboratory participation.

Each student was required to write a brief note in their report to describe their experiences working through these exercises and they also answered a brief series of questions to enable us to understand why they chose to do the exercises remotely or in the laboratory. Most students cited the convenience of being able to work on a computer that had a word processor installed -- no word processing software is installed on laboratory computers. The students who chose to do the exercises remotely (a significant majority) were roughly evenly divided between those who used on-site computer laboratories and those who used their own computers at home. When they wrote about their experiences many described how, for the first time, they found themselves free to explore the behaviour of the equipment without having to follow a laid down procedure and without having to finish everything by a certain time.

The records of their experiences and answers to an examination question for which their experience with the online laboratory provided useful background material both indicated that the learning objectives have been largely accomplished. In the lectures we explained to students that in a typical industrial situation, a simulation model is used for controller development because it would cost too much to take real process equipment off-line for several weeks of software development. In their learning experience reports, many students emphasized how convenient they found simulation environment to be when testing their controllers (because the real iron took many minutes to stabilize after transient changes), and how relatively little time most of them needed to adjust their controllers to the actual behaviour of the hardware under remote on-line control.

5 CONCLUSION

The major effect of remote access laboratory classes using the weekly exercise regime is a large increase in the time that students spend operating equipment, albeit mostly remotely. Previous studies (Lindsay and Good 2001, Tuttas *et al* 2003) have not shown significant differences in learning outcomes between remote labs and conventional labs with similar objectives. Therefore, it is unlikely that transfer effectiveness will be significantly different for each (given good pedagogic design). The significant gain in student operating time achievable with remote labs, therefore, is likely to confer significant increase in practice time and hence contribution to skill development.

ACKNOWLEDGEMENTS

The Telelabs project was supported by the Faculty of Engineering, Computing and Mathematics at the University of Western Australia. Apart from students whose work has been cited, the author would like to acknowledge contributions by Sabbia Tilli, Jan Baranski, Harjono, Icon Technologies, Paul Sorensen, and students in undergraduate classes who patiently endured unintended software faults when using remote laboratories software.

REFERENCES

AMARATUNGA, K., AND SUDARSHAN, R. (2002) *A virtual laboratory for real-time monitoring of civil engineering infrastructure*. Proceedings of International Conference on Engineering Education, Manchester, 18-21st August 2002.

BABBAGE, A. (2003) *Software System for Controlling Web-Based Laboratory Robots*, School of Mechanical Engineering Honours Thesis, University of Western Australia.

BÖHNE, A., FALTIN, N. AND WAGNER, B. (2002) *Self-directed learning and tutorial assistance in a remote laboratory*. Proceedings of Interactive Computer Aided Learning Conference, September 25-27, Villach, Austria.

DALTON, B. (2001) *Techniques for Web Telerobotics*, PhD Thesis, The University of Western Australia.

DAVIES, T. (2002) *Software system for controlling web-based laboratory*, BE Thesis, School of Mechanical Engineering, The University of Western Australia, 2002.

DEL ALAMO, J., HARDISON, J., MISHURIS, G., BROOKS, L., MCLEAN, C., CHAN, V., AND HUI, L. (2002) *Educational experiments with an online microelectronics characterization laboratory*, Proceedings of International Conference on Engineering Education, Manchester.

DEL ALAMO, J., CHANG, V., HARDISON, J., ZYCH, D., HUI, L. (2003) *An Online Microelectronics Device Characterization Laboratory with a Circuit-like User Interface*, Int Conf Engineering Education, Valencia, Spain.

FJELDY, T., AND JEPPSON, K. (2001) *Internet technology in laboratory modules for distance learning*. See <URL: http://www.unik.no/~torfj/remotelab.htm.>

GILLET, D. AND FAKAS, G., (2001) *eMersion: a New Paradigm for Web-based Training in Engineering Education*, International Conference on Engineering Education, ICEE 2001, Oslo/Bergen, Norway, August 6-10, 2001.

GOLDBERG, K., GENTNER, S., WIEGLEY, J. (1999) *The Mercury Project: A Feasibility Study for Internet Robots*, IEEE Robotics and Automation Magazine. Special Issue on Internet Robotics, December, 1999.

HENRY, J. (2003) *7 Years Experience with Web Laboratories*. Int Conf Engineering Education, Valencia, Spain. See also <URLhttp:// http://chem.engr.utc.edu/>

LINDSAY, E. AND GOOD, M. (2001) *Remote, proximal and simulated access to laboratory hardware – a pilot study*. International Conference on Engineering Education, August 6-10, Oslo, Norway.

ROSCOE, S. N. AND WILLIGES, B. H. (1980) *Measurement of Transfer of Training*, Chapter 16 of Aviation Psychology, S. N. Roscoe (Editor), Ames: Iowa State University Press. Also available on-line at <URL http://www.bluecoat.org/reports/Roscoe_80_AvPsych.pdf>

SALZMANN, CH., GILLET, D., AND HUGUENIN, P. (2000) *Introduction to real-time control using LabView with an application to distance learning*. Int. Journal of Engineering Education, 16:3. See also <URL http://iawww.epfl.ch/Staff/Christophe.Salzmann/MS_HTML/ChMS.html>

SCHÄFER, T., SEIGNEUR, J-M., DONNELLY, A. (2002) *PEARL: A generic architecture for live experiments in a remote laboratory*. Distributed Systems Group, Trinity College, Dublin. (Paper for ICSEE2003 conference, available from http://kmi.open.ac.uk/projects/pearl.

TAYLOR, K. AND TREVELYAN, J. P. (1995) *Australia's telerobot on the web*. 26th Symposium on Industrial Robotics, Singapore, October 1995, pp 39-44.
<URL:http://telerobot.mech.uwa.edu.au/>

TREVELYAN, J. P. (2002) *Towards Cost Effective On-Line Laboratories*. Proceedings of 2002 Network Learning Conference (NL2002), Berlin, May 2002 (on CD-ROM).

TREVELYAN, J. P. (2003) *Web site for Telelabs Project*, <URL:http://www.mech.uwa.edu.au/jpt/tele/>

TUTTAS, J., RÜTTERS, K., WAGNER, B. (2003) *Telepresent vs. Traditional Learning Environments – A Field Study*. Int Conf Engineering Education, Valencia, Spain.