

An Online Laboratory Framework for Control Engineering Courses*

BING DUAN, KECK-VOON LING, HABIB MIR, M. HOSSEINI and ROBERT KHENG LENG GAY
School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. E-mail: ebduan@ntu.edu.sg

In control engineering education, laboratory work is an important component for a holistic learning experience. In this paper, a survey of the e-learning related research and technologies is first presented. Then, the unique features needed for lab-based teaching and learning are highlighted. Based on this study, a lab-based learning model (LLM) is proposed to extend the e-learning framework for lab-based courses. Details of the LLM, which includes components such as the Apparatus Virtual User Interface, the Apparatus Run-time Environment and the Apparatus APIs, are discussed to illustrate how the goal of lab-based e-learning can be achieved. Finally, two example lab experiments in the teaching of control engineering courses using the proposed framework are given.

INTRODUCTION

CONTROL ENGINEERING courses conducted in the classrooms need to be integrated with design and implementation of controllers. However, setting up and preparing instructional materials for laboratory experiments is time consuming and resource intensive. There is also the limitation of space and timetable constraints to consider. Rapid progress of Internet technology and its increasing popularity have prompted several educational institutions to develop Internet- or Web-based laboratories for control engineering education [1–6].

Hitherto, most work reported concerned the setting up of remotely accessible experiment systems for research or standalone teaching purposes. From the e-learning point of view, these systems lack a learning management system (LMS) and content management system (CMS) support. Unfortunately, existing LMS does not provide a suitable framework to incorporate lab-based courses. Framework for the design, development and deployment of Internet-based lab experiments, which could interoperate with existing LMS, was seldom discussed in the open literature. A notable exception is perhaps [4]. As a consequence, these Internet-based labs are standalone and difficult to share or reuse, resulting in many functionally similar, but independent systems being developed again and again in many places. On the other hand, the e-learning community has developed several specifications (LOM [8], QTI [9], SCORM and SCO [10], etc.) to facilitate the sharing and interoperability of the e-learning materials created by different vendors and educators.

In this paper, a survey of the e-learning related research and technologies is first presented. Then,

the unique features needed for lab-based teaching and learning are highlighted. Based on this study, a lab-based learning model (LLM) is proposed to extend the e-learning framework to lab-based courses. Details of the LLM, which includes components such as the Apparatus Virtual User Interface (App-VUI), the Apparatus Run-time Environment (App Run-Time) and the Apparatus APIs (AppAPI), are discussed to illustrate how the goal of lab-based e-learning can be achieved. Finally, two example lab experiments in the teaching of control engineering courses using the proposed framework are given.

REVIEW OF E-LEARNING

The content of an e-learning system is called the learning object (LO), which is any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning (IEEE LTSC, LOM standard 2002). It is commonly viewed as the smallest element of standalone information required for an individual to achieve an enabling performance objective or outcome. For LOs to be used, they must be found. It is not easy to find anything in a large distributed online environment like the World Wide Web or a large Intranet. The solution is to store not only LOs but also *descriptions* of the LOs. Think of the LOs as data, then the descriptions are data about the data, or metadata. Thus, the basic structure of a LO can be divided into two main parts:

- the learning object itself;
- the metatag information or metadata that explains what the learning object is.

Learning object metadata (LOM) potentially includes information about the title, author,

* Accepted 26 July 2005.

version number, creation date, technical requirements and educational context and intent. Metadata can be written for an entire course, a section, a page, or even a single picture or movie clip. A huge benefit of metadata is that it can enable powerful search capabilities across content repositories. Using the object-oriented paradigm (OOP), instructional designers can build small (relative to the size of an entire course) instructional LOs that can be reused a number of times in different learning contexts. In such cases, LO becomes reusable learning object (RLO). This is a major factor in addressing the reusability aspect of the Sharable Content Object Reference Model (SCORM [10]).

Applying the RLO concept to lab-based courses, the lab-based learning model (LLM) is proposed in this paper. This LLM, which is based on the current e-learning infrastructure, serves as a framework to facilitate the design and development of lab-based learning object (LLO). The LLO is used to describe a typical lab experiment and to enhance the reusability and interoperability of the experiment created. The advantages of the proposed model are:

- sharing resources among labs/universities to provide a wide spectrum of examples;
- creating e-learning-based laboratories among different departments for a cross-disciplinary learning experience;
- allowing as many people as possible, whether commercially or from academic institutions, to contribute and develop lab-based learning objects that could be used or re-used by LMS systems.

With the proposed LLM, the lab manager can maintain the experiment resources regularly via the e-learning infrastructure. More importantly, the learner can take advantage of e-learning to carry out experiments using real instrument and apparatus from anywhere and at anytime.

To model a lab experiment as a learning object in the proposed LLM, one has to take into account features that are typical of a Web-based experiment involving real instruments or apparatus:

- The experiment consists of both the hardware and the software that controls the hardware.
- To allow users to access and control the hardware, a communication mechanism must be

provided for data exchanging between the LMS and the software that controls the hardware. This is the major difference between the LLO and the tradition LO, which only handles communication between a user and the e-learning system.

In addition, to design a searchable and reusable LLO, a suitable metadata structure, the importing and exporting and importing mechanism for the LLO must be designed.

PROPOSED MODEL

Figure 1 shows the overall architecture of the proposed LLM. The model is divided into two sections (from top to bottom):

- *SCORM-learning model*
- *App-learning model* (App is apparatus for short in this paper).

The SCORM-learning model was borrowed from the ADL [10], which deals with the launching, communicating, and tracking of content between the learning resources and the learning management system. It provides the means by which learning resources can be re-used and interoperated across multiple LMS systems.

The SCORM-learning model consists of three components:

1. *Learning resources*: represent assets (Web page, JavaScript, XML document, Flash object, picture . . .) and sharable content object (SCO) (a collection of one or more assets).
2. *LMS APIs*: the communication mechanism between LMS and SCO. These are used for collecting and logging of learning related data.
3. *SCORM run-time environment*: learning management system, which manages students and learning events to collate data on learner progress.

To extend the SCORM-learning model to the laboratory-based courses, the proposed lab-based learning model adds the following components:

1. *Apparatus virtual user interface*: or App-VUI for short, is the remote control panel for the student to control the real apparatus and observe the experiment status during the lab session.

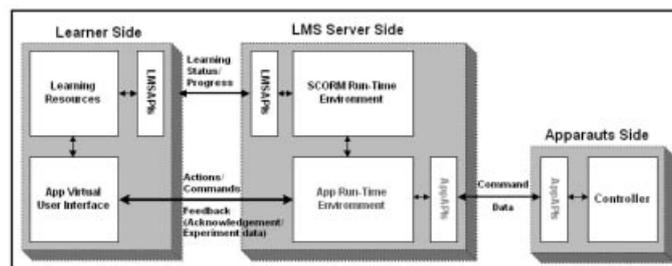


Fig. 1. Proposed lab-based learning model for e-learning.

2. *Apparatus run-time environment*: or App Run-Time for short, provides the runtime environment to manage a variety of lab apparatus and coordination with the SCORM run-time environment.
3. *Web-enabled apparatus*: consists of both the apparatus and the software that controls the apparatus.

In the proposed model, the most important element is the App Run-Time environment. It provides many standard functions such as lab session and apparatus queue management, load balancing and multi-user collaboration. These functions are reusable and shared by all LLOs, which are the experiments in the lab-based courses.

COMPONENTS DETAIL

App Run-Time environment

As discussed before, the App Run-Time environment is the key to achieve the e-learning goal in lab-based courses. Figure 2 depicts the proposed App Run-Time environment model.

There are three types of interface for the App Run-Time to communicate with other modules:

- On the left side of Fig. 2, the message processor is responsible for interacting with the App-VUI. It receives the control commands and feedback the experiment data and status to the remote user.
- On the right side of Fig 2, the apparatus application programming interfaces (or App APIs for short) are responsible for the communication between the App Run-Time environment and Web-enabled apparatus. These APIs provide all the necessary functions for initiating a connection and delivering data and control commands to the real apparatus.
- At the top of Fig. 2, the information about the learner and learning progress during a laboratory session is communicated to the SCORM Run-Time environment through the LMS interface.

The Run-Time environment can be considered as a container (or framework) for lab-based learning objects. Every LLO integrated into the environ-

ment can then share the resources provided by the framework. Typical resources provided by the framework are: lab session and apparatus queue management, load balancing, experiment report generator, communication services with physical apparatus, live video, etc.

App-virtual user interface

Interactivity is essential to carry out an experiment from a remote Web-browser. The App-virtual user interface (App-VUI) provides the remote control panel, which enables the student to issue control commands to the apparatus remotely. Meanwhile, the results of the experiment, such as the response and status of the apparatus, are all shown in the App-VUI display area. To start a lab session, the App-VUI sets up a HTTP connection with the App Run-Time environment via simple object access protocol (SOAP). Once the connection is authorized, the student starts to conduct the experiment. Data transmitted over the Internet will be delayed depending on the communication bandwidth available and adopted protocol [7]. To design a reliable App-VUI, the Internet time delay issue needs to be considered.

Here, we adopt a protocol, which requires the server to send an acknowledgement message back to the client to confirm that the control command from the client has been received.

Web-enabled apparatus

Basically, the lab experiments are treated as a black box in the online laboratory system. These black boxes just provide the interface to communicate with the e-learning system. All data, i.e., apparatus status, experiment feedback, are transferred via the interface.

Figure 3 depicts a Web-enabled apparatus model, which is a combination of the hardware and software to fulfil a specified experiment task.

Typically, a real hardware (e.g., electrical device, medical equipment, optical experiment) is connected with a controller, which supports the TCP/IP function. The controller can be an embedded processor or PC. For the software part, XML parser and packager can be used to process the control commands, experiment status as well as the raw data. As a whole unit, the

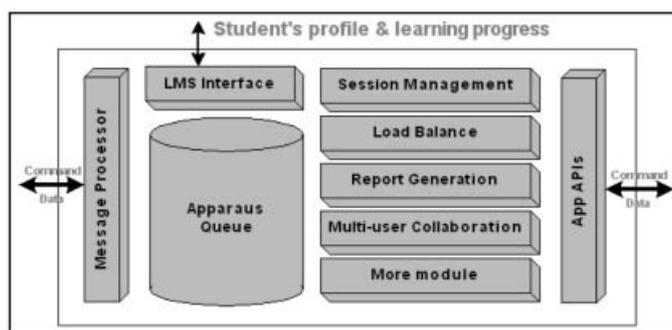


Fig. 2. Proposed App Run-Time environment model.

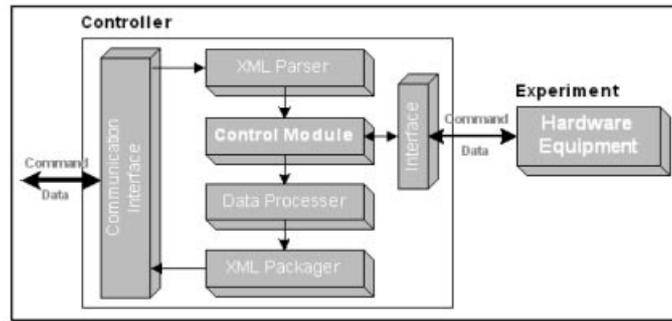


Fig. 3. Proposed Web-enabled apparatus model

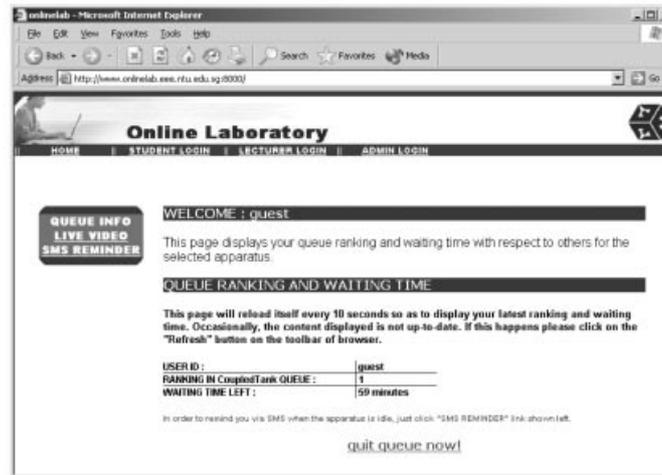


Fig. 4. Apparatus queue with timeout feature.

controller must have the necessary interface, such as DA/AD converter, data acquisition card to link with the actual equipment.

Under normal conditions, the controller listens to the request from a TCP port. Once a command is received, verification begins to make sure that the format and value of command are correct. Otherwise, the command will be discarded. In both cases, an acknowledgement will be sent back to the requester. Because the communication is based on socket connection, which is platform independent, the apparatus can work in different e-learning systems and be reused in many learning contexts. Control parameters, apparatus responses and experiment data are encoded into XML-based messages, which can be easily customized.

R. Pastor [4] has presented an XML-based framework, **REmote Laboratory Extended (RELATED)**, for the development of Web-based laboratories. The idea is to define an abstract entity called RLAB system by an XML DTD so that lab experiments can be described by RLAB. Then, the RLAB schema is published on the server for general use. The same idea is introduced in the model. The XML is used to describe the structure and attributes of Web-enabled apparatus. Hence, it can be easily recognized and identified by the e-learning LMS or CMS. The changing, modifying or upgrading of the experiment is done by edit the XML schema.

Based on the proposed model, personalized learning objects, which can provide different learning contents or sequences to the learner according to their learning experiences, can be achieved.

EXAMPLE APPLICATIONS

In this section, we show two examples for control engineering courses developed based on the proposed LLO concept and also an example of the App Run-Time environment.

App Run-Time environment

Considering a real laboratory, the lab technician manages the lab and students are assigned to use certain lab apparatus and to carry out experiments in the lab according to a timetable. Similarly, in the online laboratory, the App Run-Time environment will provide the necessary management of the 'Internet Laboratory'.

Figure 4 shows one of the common services that the Run-Time environment will support, which is a first-in-first-out apparatus queue with a timeout feature. This is to ensure that only one remote user is allowed to have control of the apparatus at any one time, and the timeout feature prevents a user from hogging the apparatus.

The Run-Time environment also provides functions for a lab technician to remotely manage the

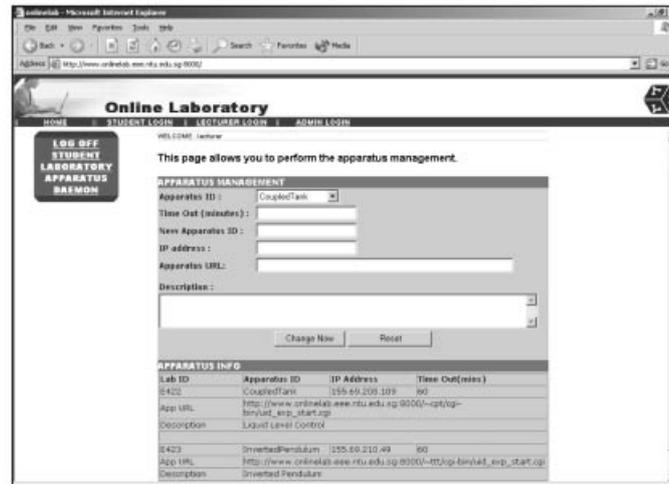


Fig. 5. Remote management of apparatus.

lab apparatus (see Fig.5). Examples of such functions are add new experiment and apparatus to the e-learning course and take an apparatus offline for maintenance.

The rotary inverted pendulum

The rotary inverted pendulum (see Fig. 6) is an apparatus widely used for the teaching of advanced control system theory in the laboratory. The angle of the pendulum is measured with a potentiometer, while the speed of the motor is measured with an encoder. There is a built-in microcontroller board on the inverted pendulum apparatus. It contains a microcontroller, ROM, RAM, timer, serial ports, PWM outputs as well as A/D inputs. The challenge is to swing up the pendulum from its downward position and balance it upright. It is often used as a testbed to demonstrate the performance of difference control algorithms, from classical controllers to modern neural-fuzzy controllers.

One example of the lab-based learning object for the inverted pendulum shown in Fig. 7 is the system identification module. It is used to estimate

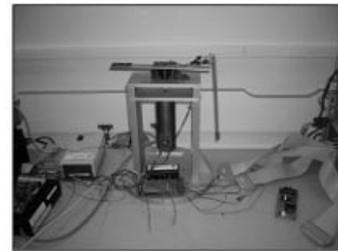


Fig. 6. Inverted pendulum.

the parameters that characterize the dynamics of the inverted pendulum and its motor. These parameters are required for the controller design. Through the Web interface, students set the magnitude and the input test signal. Once the 'Start' button is clicked, the pendulum will start to spin for a short duration and the values of the pendulum position and the arm velocity will be saved into a data file. It is used to perform the necessary computations to determine the pendulum parameters based on the experimental data collected.

Students could then download the data file and

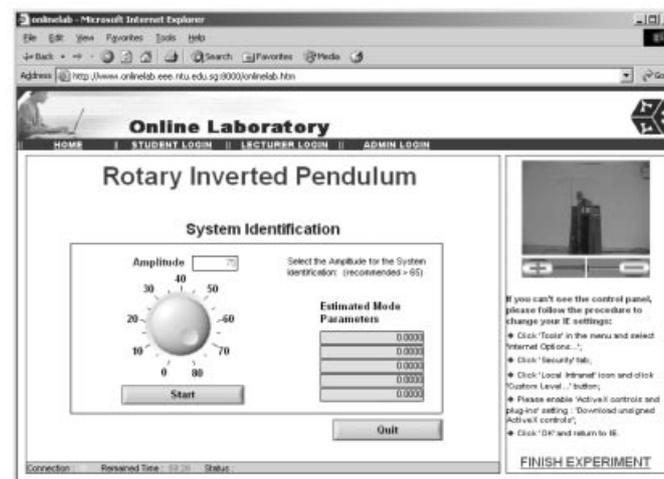


Fig. 7. Inverted pendulum system identification.

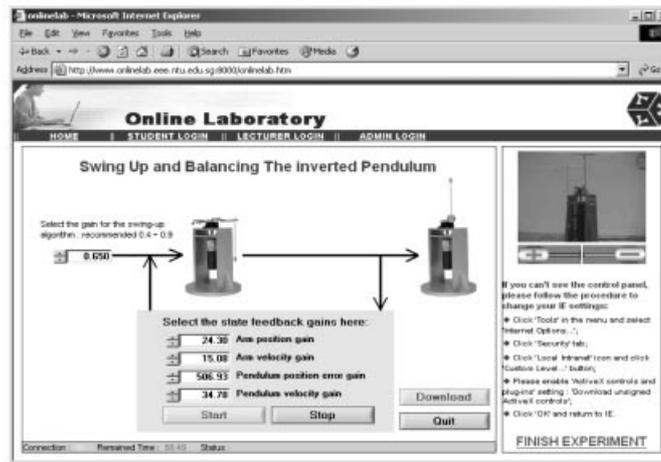


Fig. 8. Swinging and balancing the pendulum.



Fig. 9. Coupled tank.

perform the system identification calculation using their own program or Matlab script.

Figure 8 shows another LLO, which is the swing and balance module. It lets students specify the controller parameters, which they have designed, and then starts the controller to investigate if his/her design would be able to balance the pendulum on the actual inverted pendulum. Students can refine his/her design and test the controller experimentally. This example also demonstrates that

more than one LLO could be designed for the same hardware apparatus.

From the live video feedback provided by the LLM, students could confirm whether the apparatus is in working order during the experiment session. Hence, the students can learn from this experience and improve their controller design skills using a real apparatus.

The coupled tank

The coupled tank apparatus shown in Fig. 9 consists of two small Perspex tower-type tanks mounted above a reservoir which functions as storage for the water. Two independent pumps pump water into the top of each tank. The apparatus is designed for teaching elementary feedback control principles.

A FieldPoint (FP2000) running LabVIEW (Laboratory Virtual Instruments Engineering Workbench) functions as the controller for the coupled-tank apparatus. The objective of this experiment is to maintain the water levels in the tanks at the specified heights.

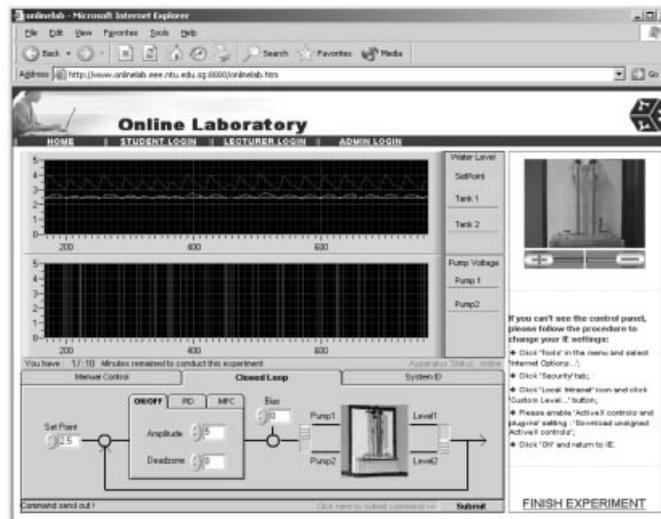


Fig. 10. Coupled tank App-VUI.

The student interacts with the apparatus through a customized App-VUI of the Coupled Tank shown in Fig. 10. This App-VUI allows the student to experiment with different controllers, such as manual, ON/OFF, PID (proportion-integral-derivative) control, and to study the effect of adjusting the various parameter settings available in the controller. As this is a relatively slow process, the water levels and the control signals to the pump can be charted on the App-VUI as the experiment progresses.

The student can also download all data collected by the LLM for further analysis when the experiment has been completed. With these applications developed, the student can now have hands-on sessions for their control classes and acquire a better appreciation of the control theories taught in the classroom.

CONCLUSIONS

In this paper, a lab-based learning model, which can be integrated with the current e-learning system, has been proposed. It has been shown that the proposed model could facilitate the design and deployment of the lab-based courses. Details of the proposed model, the design of Apparatus Virtual User Interface, Apparatus Run-time Environment and Apparatus Application Program Interfaces for laboratory experiments were also discussed and examples of lab-based e-learning modules for control engineering education were presented. There are still many issues to be resolved before a full-fledge lab-based e-learning system can become a reality. The possibility of providing lab-based e-learning using grid computing techniques should be explored.

REFERENCES

1. K. V. Ling, Y. Lai and K. Chew, An online Internet laboratory for control experiments, in *Advances in Control Education*, Pergamon, UK (2000) pp. 173–176.
2. Kok Kiong Tan, Tong Heng Lee and Chai Yee Soh, Internet-based monitoring of distributed control system—an undergraduate experiment, *IEEE Trans. Education*, **45**(2), 2002, pp. 128–134.
3. Chi Chung Ko, B. M. Chen, Shaoyan Hu, V. Ramakrishnan, Chang Dong Cheng, Yuan Zhuang and Jianping Chen, A Web-based virtual laboratory on a frequency modulation experiment, *IEEE Trans. Systems, Part C, Man and Cybernetics*, **31**(3), 2001, pp. 295–303.
4. R. Pastor, J. Sanchez and S. Dormido, An XML-based framework for the development of web-based laboratories focused on control systems education, *Int. J. Eng. Educ.*, **19**(3), 2003, pp. 445–454.
5. S. H. Yang, X. Chen, J. L. Alty, Design issues and implementation of internet-based process control systems, *Control Engineering Practice*, **11**, 2002, pp. 709–720.
6. C. C. Ko, B. M. Chen, Chen Jianping, Y. Zhuang and K. Chen Tan, Development of a Web-based laboratory for control experiments on a coupled tank apparatus, *IEEE Trans. Education*, **44**(1), 2001, pp. 76–86.
7. Kuk-Hyun Han, Sinn Kim, Yong-Jae Kim and Jong-Hwan Kim, Internet control architecture for internet-based personal robot, *Autonomous Robots Journal*, **10**(2), 2001, Kluwer Academic Publishers, pp. 135–147.
8. IEEE Learning Technology Standards Committee (LTSC), <http://ltsc.ieee.org>
9. IMG Global Learning Consortium (IMG), <http://www.imglobal.org>
10. Advanced Distributed Learning (ADL), <http://www.adlnet.org>

Duan Bing received his B.Eng in School of Communication Engineering from Chong Qing University (China) in 1999. He was a database administrator from 1999 to 2001 in China Telecom and software engineer in Singapore. He is now pursuing his master degree in ICIS, School of Electrical Electronic Engineering, Nanyang Technological University of Singapore. His major research interests are focused on e-learning framework for lab-based subjects, XML-based instrumentation markup language and network traffic analysis for real-time control.

Ling Keck Voon obtained his D.Phil. from University of Oxford, United Kingdom and B.Eng. from National University of Singapore. He is currently an Associate Professor at the School of Electrical & Electronic Engineering of Nanyang Technological University, Singapore. His research interest is in control and automation.

Habib Mir Hosseini received his B.Sc. in electronic engineering from Isfahan University of Technology (Iran) in 1988, his M.Sc. in digital electronics from Sharif University of Technology (Iran) in 1991, and his Ph.D. in electrical engineering from the Adelaide University (South Australia) in 1998. He joined Iranian Telecommunication Research Centre (ITRC) as a design engineer in 1988. In 1992, he joined Behineh Pardaz Company as software engineer. From December 1997 he was a research fellow with the Network

Technology Research Centre (NTRC), School of EEE, Nanyang Technological University of Singapore, and he later joined the School of Electrical and Electrical Engineering of NTU in August 2000 as an Assistant Professor. His research interests include multimedia encryption and copyright protection (digital watermarking), pattern recognition, e-learning models and document understanding and recognition.

Robert Kheng Leng Gay obtained his B.Eng. (1965), M.Eng. (1967), and Ph.D. (1970) in electronics engineering, from the University of Sheffield. Prof. Gay is the director of the IT Research Infrastructure, also director and CEO of the Managed Computing Competency Centre (MC3) of Nanyang Technological University, Singapore. He was a member of the team that was awarded the LEAD Award for CIM (Leadership, Education and Application in Development of Computer Integrated Manufacturing) by the Society of Manufacturing Engineers, USA in 1992. His recent research interests are focused on advanced knowledge-based systems, e-learning, agent-based technology and industrial IT.