Web-Based 3-D Control Laboratory for Remote Real-Time Experimentation

Wenshan Hu, Guo-Ping Liu, *Fellow, IEEE*, and Hong Zhou

***Abstract*—The design and implementation of Networked Control System Laboratory (NCSLab) 3-D which is a web-based 3-D control laboratory for remote real-time experimentation are introduced in this paper. NCSLab 3-D is built based on the NCSLab framework which supports the structure that the test rigs are located diversely in different parts of the world. In NCSLab 3-D, the test rigs are cataloged into several sublaboratories according to their functionalities. The laboratory building, sublaboratories, and test rigs are modeled in 3-D and reconstructed in a web-based interface using Flash 3-D engines. Users can "walk into" these laboratories and pick up the test rigs in a virtual reality environment similar to what they do in hands-on laboratories. During the remote experiments, the 3-D models are synchronized with the real test rigs through the network data links. Users are able to zoom in, zoom out, and rotate the 3-D models freely. Therefore, the real-time experiments can be watched from any angles. NCSLab 3-D has been applied to the control engineering education in Wuhan University, China. The results of the teaching practice show that NCSLab 3-D are able to bring great convenience to both users and maintenance personnel and improve the efficiency of the laboratory equipment significantly.**

***Index Terms*—Control engineering education, remote laboratories, 3-D virtual reality, web-based applications.**

I. Introduction

With the advances in network and web technologies, online laboratories, including remote laboratory [1], [2], virtual laboratory, and hybrid laboratory [3], have developed rapidly in recent years. A remote laboratory is a set of experiments using real equipment in a physically "live" laboratory, enhanced by software applications and technologies, connected to the Internet and enabling remote access to real setups and their devices [4], [5]. Even though remote laboratories are more complex and more expensive than simulation-based virtual ones, a chance of working with real instead of simulated data may induce higher level of motivation for students [6].

Manuscript received March 9, 2012; revised May 21, 2012; accepted June 18, 2012. Date of publication July 27, 2012; date of current version May 16, 2013. The work of W. Hu was supported by the National Science Foundation of China under Grant 61004030. The work of G.-P. Liu was supported in part by the National Natural Science Foundation of China under Grants 61028010 and 61021002. The work of H. Zhou was supported by the Fundamental Research Funds for the Central Universities of China under Grant 208275642.

W. Hu and H. Zhou are with the Department of Automation, Wuhan University, Wuhan 430072, China (e-mail: wenshan.hu@whu.edu.cn; hzhouwuhee@263.net).

G.-P. Liu is with the Faculty of Advanced Technology, University of Glamorgan, CF37 1DL Pontypridd, U.K., and also with the Center for Control Theory and Guidance Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: gpliu@glam.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2012.2208440

Remote laboratory technology connects the limited experimental devices with the Internet, which makes it possible for distance experimentation and enables the share of laboratory resources between educational institutions without the restrictions of the geographical locations [7]. The efficiency of these resources can be boosted greatly by the remote experimentation technology.

In the area of automatic control, there are many systems implemented successfully, such as dc servo motors [8], inverted pendulums [9], robots [10], [11], ac electric machines [12], 2-DOF helicopters [13], and programmable logic controllers (PLCs) [14]. Various technologies and frameworks [15]–[20] have been adopted in the development of remote laboratories, such as Matlab [21], [22], Java [9], [23], Asynchronous JavaScript and XML (AJAX) [24], iLab [25], LabView [26], etc.

Complex architectures beyond the traditional two-layer server–client structure [27] are also adopted to extend the accessibility of remote laboratories. WebLab-Deusto [28] enables different kinds of test rigs located diversely in different institutions to integrate under a single web interface. In [29], different PLCs, together with several types of sensors, actuators, and industrial communication networks, are integrated in a remote automation laboratory using a cluster of virtual machines.

Laboratories such as the Automatic Control Telelab [30] and the University of Alaska Fairbanks Remote Robotics and Control Laboratory support and even encourage users to design their own control algorithms and customize the reference inputs. In [31], an intelligent help system is developed to support the student to conduct remote experimentation. A collaborative remote experimentation environment is introduced in [32]. These systems give great flexibility and convenience to students.

Networked Control System Laboratory (NCSLab) [33]–[35] provides a unified and flexible web-based interface [36] for various test rigs such as servo motors, magnetic levitations, etc. It is able to combine various experiment devices located globally and offer various experimentation services for users scattering around the world. Users can use the web browser in their PC systems to design control algorithms and customize their monitoring during experiments without requiring the knowledge of programming. The NCSLab was initially developed in the U.K., and it has been gradually implemented in several institutions located in both U.K. and China.

II. Three-Dimensional Virtual Reality for Remote Laboratories

A survey conducted in [37] indicates that, despite the convenience and reliability of remote laboratories, students still

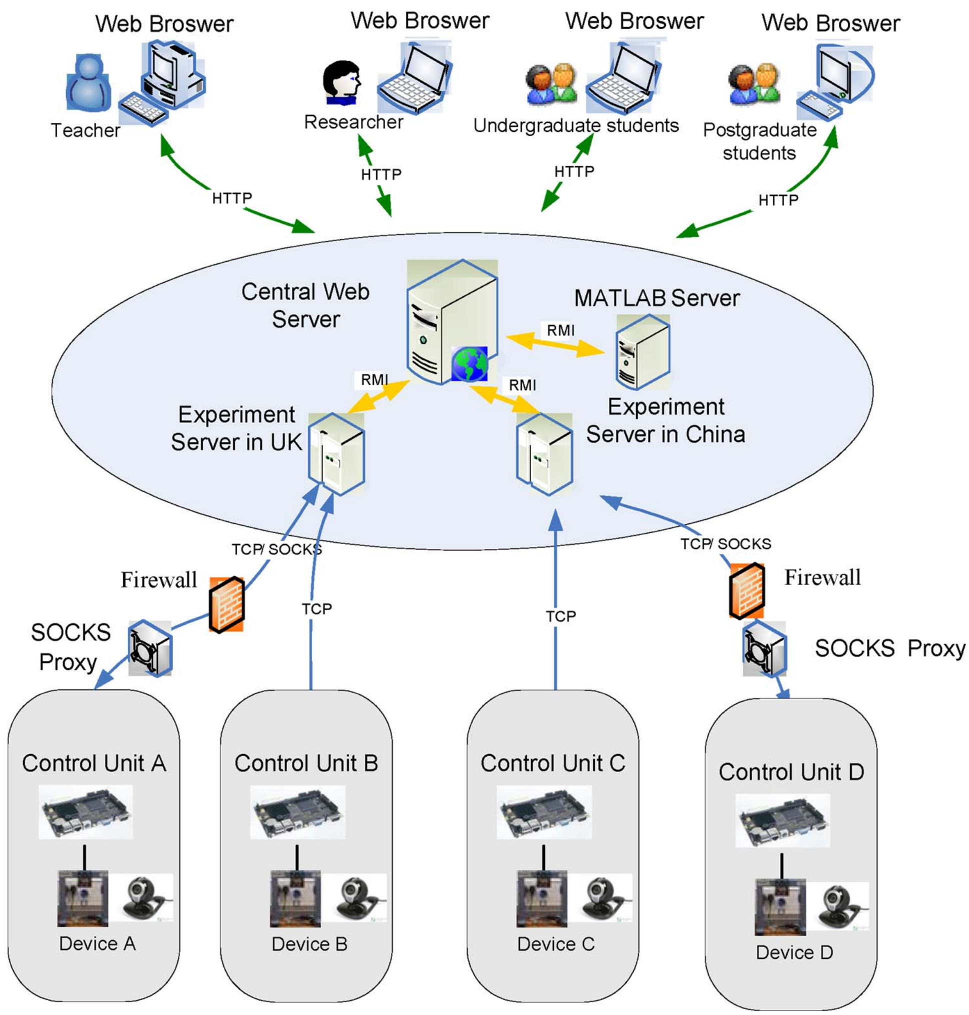


Fig. 1. Architecture of NCSLab.

expressed preference for hands-on ones. Without face-to-face operation, remote laboratories are not able to bring equal sense of immersion to users comparing with hands-on ones.

In order to provide users a more vivid laboratory experience in remote mechatronic experiments, 3-D and virtual reality technologies [38]–[41] have been used as appealing solutions in remote laboratories [42], [43]. SecondLab [44] which works over WebLab-Deusto allows students to control microbots from Second Life and gives them the chance to work with real experiments from a social 3-D-based immersive environment. A remote laboratory in an Open Virtual world using Java Web Start is reported in [27]. The design of game-based online laboratories with 3-D virtual environment is investigated in [45] and [46]. A methodology for remote experimentation in a virtual 3-D world with a sense of real presence and interaction is proposed in [47].

There are several solutions available for web-based 3-D virtual realities. Web Graphics Library (WebGL) which uses the HTML5 canvas elements is a set of JavaScript Application Programming Interfaces (APIs) for rendering interactive 3-D graphics within any compatible web browsers. Google Native Client (NaCl) provides a set of 3-D APIs for web-based 3-D rendering. Binary Format for Scenes (BIFS) under MPEG-4 standard is another solution to represent 3-D scenes and objects in web-based applications [48].

Stage 3-D which is a part of Flash 11 is a new method in 3-D rendering. It supports a set of low-level graphic processing unit (GPU)-accelerated APIs. For the convenience of the developers, there are also a few 3-D engines based on Stage 3-D available, and most of them are open source software. Even though the expansion of WebGL, NaCl, and BIFS has been very fast in recent years, Adobe Flash still has the biggest market share and has been installed in more than 95% of Internet-enabled PCs so far, according to a survey conducted by Millward Brown in July 2011, which is high above other competitors.

In this paper, NCSLab 3-D which is a 3-D web-based control laboratory based on the NCSLab framework will be presented. NCSLab 3-D adopts Flash as the 3-D graphic engine, and therefore, it is compatible to most web browsers due to the popularity of Flash. Laboratory buildings, rooms, and test rigs are modeled in 3-D and reconstructed in web browsers. Users can “walk” into the 3-D laboratory rooms and select test rigs similar to what they do in hands-on laboratories. When users are doing experiments, 3-D replicas of the test rigs are brought to them as well as the real-time videos. The 3-D models are synchronized with the real test rigs so that users are able to "monitor" experimental processes through the animations in real time.

III. Background and 3-D Demands of NCSLab

A. Background of NCSLab

The proposed 3-D web-based laboratory is based on the NCSLab framework, and therefore, it is necessary to introduce some background information about NCSLab. To support the global scale web-based experimentations, NCSLab has a six-tier architecture which is shown in Fig. 1. From the top–down view, the architecture of NCSLab consists of web browsers, central web server, Matlab servers, regional experiment servers, SOCKS, proxy servers, device control units, and experiment devices.

The complex architecture enables reliable communication channels among the servers, web browsers, and control units diversely located in different parts of the world. Users can have access to all the test rigs integrated in NCSLab under a universal web-based interface without considering their physical locations. The servers collect the real-time data from the test rigs and distribute these data to end users' web browsers. The web browsers display these data in a web-based monitoring interface with many 2-D widgets like dynamical charts, sliders, numeric inputs, etc. The features of the NCSLab are listed as follows.

1) *Geographically diversely located test rigs*: The test rigs in NCSLab can be located diversely in different parts of the world.
2) *Sublaboratory configuration*: All the test rigs are classified by their functionalities. For example, all the flow control test rigs are put into the Flow Control Sublaboratory, and all the servo control test rigs are cataloged in the Servo Control Sublaboratory no matter where their geographical locations are. Users can pick up any test rigs from the corresponding sublaboratory without knowing its real location.
3) *Visual configuration of monitoring interface*: NCSLab also gives users the freedom to customize their own monitoring interface, in which they can choose the signals to be observed and the parameters to be tuned during experiments.
4) *Web-based control algorithm design*: The users are encouraged to design their own control algorithms. There are two methods provided in NCSLab. First, the users can use Matlab/Simulink to customize the algorithms and then upload them to NCSLab using a web-based interface. Second, NCSLab also provides a pure web-based GUI called WebconLink which enables users to design their algorithms by drawing block diagrams in their web browsers.

B. Three-Dimensional Demands of NCSLab

NCSLab has been implemented in several institutions both in U.K. and China since 2006. It has been applied in many teaching modules such as *Control Theory, System Identification, Process Control, etc*. Through these teaching practices, a lot of valuable feedbacks have been obtained, which encourages the further development of the system.

Fig. 2. Welcome page of NCSLab 3-D.

The questionnaire embedded in the NCSLab Web site indicates that 83.3% of the users had chosen "satisfied" or "very satisfied" when they were asked if they were satisfied with the remote experiments on NCSLab in general. However, when they answered the question if they believed they controlled the actual experiment apparatus when they were doing experiments on NCSLab, only 36.7% of them chose "firmly believe," and 56.7% chose "partially believe." These feedbacks indicate that, although users can do most of the jobs in NCSLab as they do in hands-on laboratories, they are not able to get quite the same laboratory experience as they do in hands-on ones.

To address the problem, NCSLab has been modified and web-based 3-D technologies have been adopted in the new version. Fig. 2 shows the welcome page of the NCSLab 3-D. There are two targets for the new development. First, all the 3-D virtual realities should be implemented in the web browser. Second, as a feature of the NCSLab, the 3-D technology should not require any special plug-in. Users are able to have access to the laboratories in web browsers without requiring the installation of any special software.

IV. Design of Web-Based 3-D Laboratory

In order to display the 3-D models in the web-based interface of NCSLab, various supporting technologies are integrated in the system design.

A. Three-Dimensional Modeling

The components (test rigs, laboratory rooms, and buildings) of the NCSLab 3-D must be modeled in 3-D before it can be imported into the web-based user interface. There are many commercial software available for the 3-D model design, such as 3DS Max, Solid Works, Pro/E, etc. The original 3-D models to be displayed in the web-based interface were firstly designed to use the software and then converted into a common format which could be recognized and decoded by Flash 3-D engines.

In the case of NCSLab 3-D, 3DS Max is selected as the 3-D model development tool. The models designed in the 3DS Max are exported into the Wavefront.obj format for the Flash 3-D engine. Fig. 3 is an example which shows the 3-D model of a dual-tank test rig being designed in the 3DS Max environment.

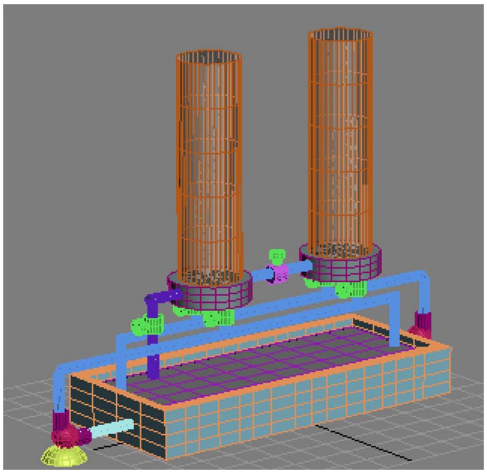


Fig. 3. Three-dimensional model of a dual-tank test rig in 3DS Max.

When designing the 3-D models, the complexity must be considered carefully. The quality and the complexity of the 3-D models have to be balanced. Even though the most powerful Flash 3-D engines are more than capable of processing tens of thousands of triangles, too complex models always result in big target files. In the Internet environment, the long wait for downloading these files to the web browsers should be avoided as much as possible.

B. *Flash 3-D Engine*

Three-dimensional models cannot be displayed in the web browser directly. They must be imported and rendered in Flash Controls. There are several Flash 3-D engines available. Paperverion3D, Away3D, and Sandy3D are the examples. However, some early versions of 3-D engines only support software rendering. Without GPU acceleration, only relatively simple 3-D applications can be implemented in a web-based interface, which have significantly limited the expansion of the web-based 3-D applications.

The release of Flash 11 has changed the situation. Flash 11 starts to support hardware-accelerated 3-D rendering, which makes it possible to design more complex 3-D applications. Stage 3-D, which is a set of 3-D APIs in Flash 11, enables advanced 3-D capabilities on both PC and mobile platforms. In order to help developers to design 3-D web-based applications quickly, Proscenium, which is an ActionScript code library built on the top of Stage 3-D, has also been released. By using Proscenium, models designed using 3-D design software can be easily imported and rendered into Flash Controls. The structure of 3-D Flash Controls in the NCSLab is shown in Fig. 4.

By using the resources provided by Stage 3-D and Proscenium, 3-D models designed in the 3DS Max environment are imported into Flash Controls, which are embedded into the web browser. Therefore, they can be embedded and displayed in the web-based interface. Fig. 5 shows the dual-tank test rig rendered in a 3-D Flash Control. Apart from the 3-D engine, some supporting codes have also been designed. These codes are able to communicate with the Web Servers and dynamically control the animations of the 3-D components.

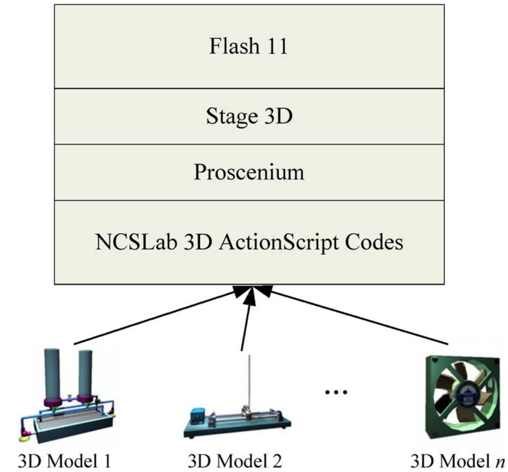


Fig. 4. Structure of a 3-D Flash Control.

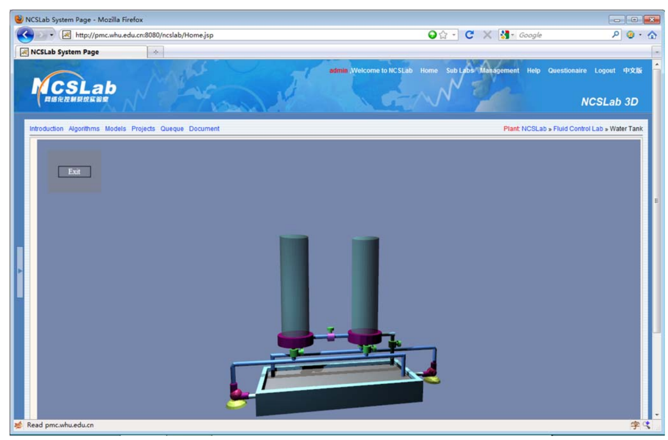


Fig. 5. Dual-tank test rig in NCSLab 3-D.

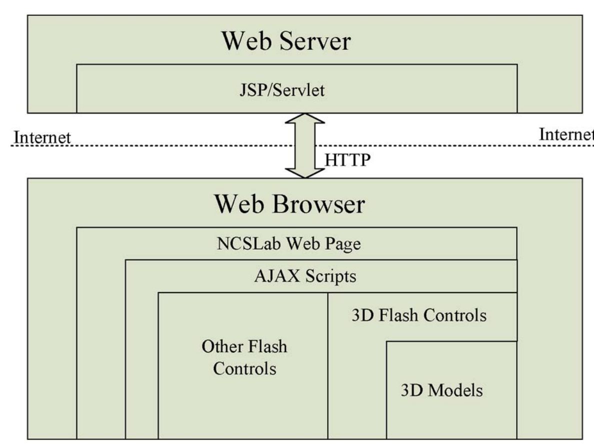


Fig. 6. Structure of 3-D web-based interface.

C. *Web-Based 3-D Interface*

Fig. 6 shows the structure about how the 3-D models are displayed in the web-based interface of NCSLab. The Tomcat Web Servers dynamically create the NCSLab web pages using java server pages (JSP)/Servlet Technology. These pages are downloaded to the web browsers and generate the web-based interface.

On the user's web browser, AJAX scripts embedded in the HTML codes are designed to deploy the 3-D Flash Controls in

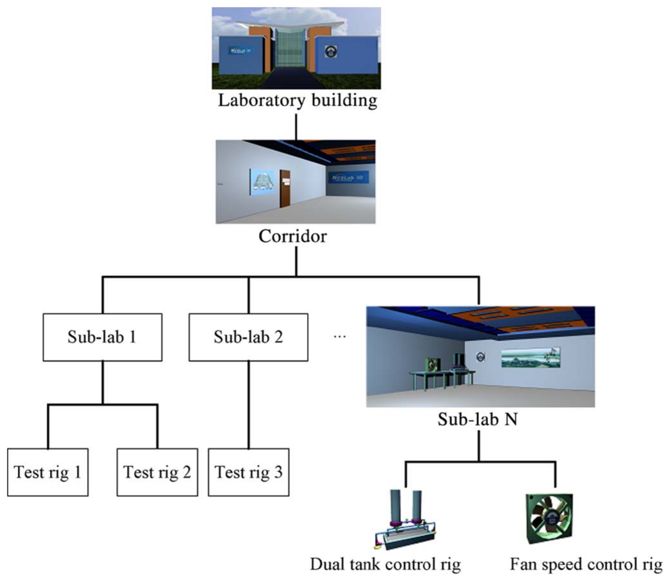


Fig. 7. Logical structure of the laboratory building.

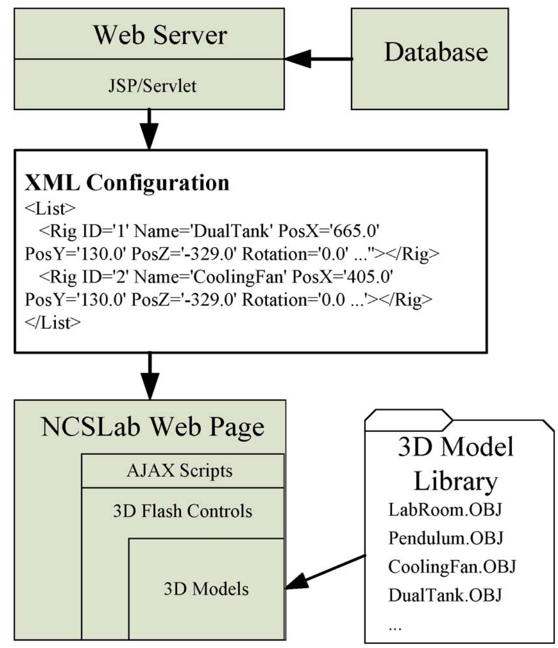


Fig. 8. Three-dimensional sublaboratory configuration.

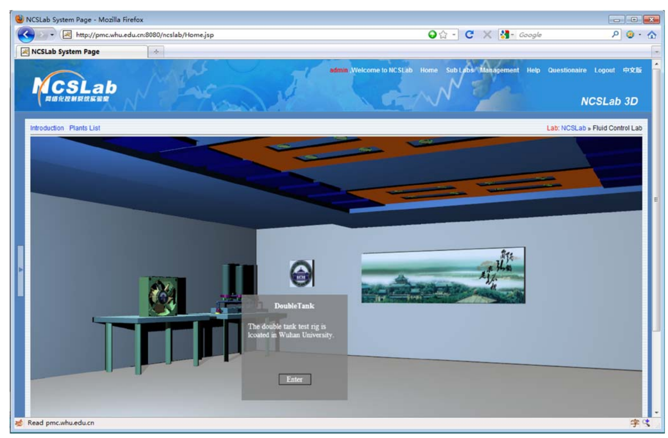


Fig. 9. Three-dimensional sublaboratory in NCSLab 3-D.

the web-based interface. These codes download the corresponding 3-D Flash Controls and embed them into the web browser. The 3-D Flash Controls are also able to obtain real-time data from the web server. The motions of the 3-D models are manipulated according to the real-time experimental data collected from the controller through the web server, which gives the synchronization between the 3-D models and real test rigs.

Even the same 3-D Flash Controls running in different circumstances may have different configurations. These configurations are generated dynamically in the extensible markup language (XML) form by the JSP/Servlet codes running on the web servers. The AJAX scripts transmit the XMLs to the 3-D Flash Controls where the 3-D components are displayed properly according to these configurations.

V. Implementation of NCSLab 3-D

A. Structure of the Laboratory Building in the Virtual Reality World

In the NCSLab, all the test rigs can be cataloged into several sublaboratories according to their functionalities rather than locations. Sublaboratories are put into a laboratory building which only exists in the web-based interface.

The logical structure is shown in Fig. 7. When users log in in the NCSLab 3-D, the 3-D laboratory building is displayed in their web browsers. They can use mouse and keyboard to "walk into" the virtual building.

Inside the building, the users step into a corridor with many doors. Each door is the entrance to a certain sublaboratory room. The names of the laboratories are written on the doors. By clicking the mouse on these doors, the user can check the detailed information of the corresponding sublaboratory and choose which room he would like to enter.

B. Three-Dimensional Sublaboratories

Fig. 8 shows an example of how a 3-D sublaboratory room is displayed in the web browser. Even though the sublaboratory rooms only exist in the virtual interface, the test rigs are all modeled with real dimensions. The AJAX scripts running in the NCSLab web pages get an XML configuration from the web server. The test rigs in the sublaboratory are listed in the XML, and their detailed information such as the identity (ID), model name, and position is also specified. The XML is configured in advance and stored in the MySQL database on the server side. The 3-D Flash Control parses the received XML, loads the 3-D models of the laboratory room and test rigs according to the information obtained from the XML, and sets up the laboratory room in the web-based 3-D environment.

Fig. 9 shows a 3-D sublaboratory which has two test rigs. Users can pick up any test rig available in the virtual world as they point the mouse cursor to the test rig and press the left button. After the choice has been made, the browser will be redirected automatically to the chosen test rig in the experimental web page.

C. Communication Channels Established in Remote Experiments

When users enter the web page for the experiments on the chosen test rig, they can design new control algorithms,

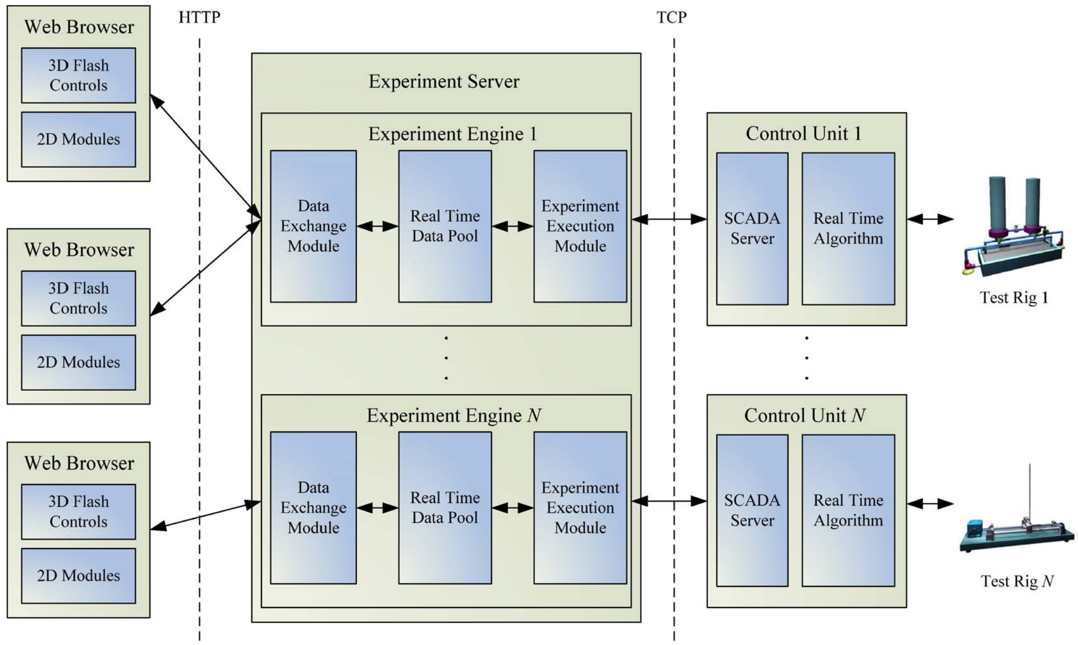


Fig. 10. Structure of the data link.

download the algorithms to the target Control Unit, and monitor how the experiments are going on in real time in the NCSLab web-based interface. In the monitoring and supervising interface, they have two choices to "observe" the test rigs. One is from the real-time video collected from the cameras installed near the test rig, and the other is from the 3-D real-time animations which are created in the web-based interface.

Communication channels are established for the data exchange between the web browsers and Control Units during the remote experiments as shown in Fig. 10. The data link has two tasks. One is to establish the synchronization between the widgets (both 3-D models and 2-D modules) in the web browser and real test rigs. The other is to pass the users' remote instructions (tuning parameters, changing algorithms, etc.) to the Control Unit.

For each test rig, there is a corresponding Experiment Engine deployed in one of the Experiment Servers. The Experiment Execution Module is the interface to the test rig. When a control algorithm is downloaded and executed in the Control Unit, the transfer control protocol communication is established automatically between the Experiment Execution Module and the Supervisory Control and Data Acquisition Server running in the Control Unit. It collects the real-time data from the real test and temporarily stores them into a Real-Time Data Pool.

When a user starts to monitor the experimental process, the web browser downloads the web pages, widgets (3-D Flash Controls and 2-D Modules), and 3-D models from the Experiment Server. After the downloading process has been finished, the hypertext transfer protocol (HTTP) communication between the widgets and the corresponding Data Exchange Module is created automatically. The data from the Real-Time Data Pool are transmitted to the web browser and delivered to the widgets through the HTTP connections.

The Real-Time Data Pool is a buffer between the web browsers and Control Unit. It has a first-in-first-out structure and stores the temporary real-time data in case of network delay. When the network condition is good (for campus network), the real-time data collected from the Control Unit can be fetched quickly by the web browser. If the network work condition is not perfect (for long-distance Internet transmission), a small time delay is inevitable, but the continuality of the data stream between the web browser and Web Server can still be guaranteed.

Apart from the 3-D test rig models, there are also some widgets like sliders available for users to tune up the control parameters. When the value of a parameter is changed by the user, the corresponding instruction is passed to the Data Exchange Module as an HTTP request. The Data Exchange Module sends the instruction to the Experiment Execution Module where the instruction is further passed to the Control Unit.

D. Three-Dimensional Animation Synchronized With the Test Rig

In the Proscenium library, a callback function called onAnimate() is defined to manipulate the animations. Developers can override the function to control the movement of the component in the 3-D Flash Controls. There are also various motion controls that APIs provide. The 3-D components can be easily moved, rotated, and scaled by using APIs.

For a simple example, if a user is working on a dual-tank flow control test rig through NCSLab 3-D, the real-time data of the water levels for both the left- and right-hand-side tanks are passed to the 3-D Flash Controls in the user's web browser via the data link. The 3-D Flash Control changes the water levels on the 3-D models according to the data that it receives, where the synchronization is established.

The 3-D animation for the dual tank is achieved by the following way. When the 3-D Flash Control initiates, only the

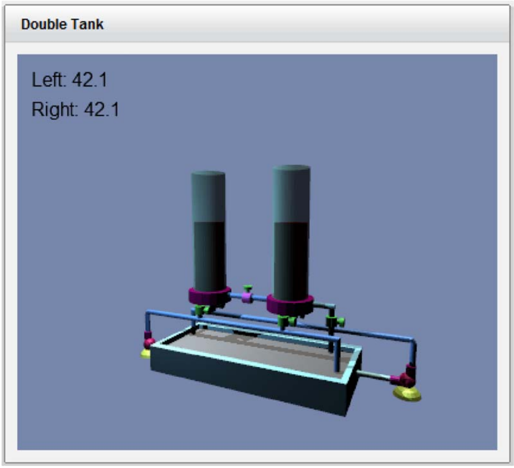


Fig. 11. Three-dimensional animation of a dual-tank test rig in NCSLab 3-D.

model with two empty and transparent tanks is loaded as shown in Fig. 5. Two cylinders which represent water are created and placed inside the two tanks. During the experiment, the heights of the two water cylinders are modified dynamically according to the real-time data from the web server. From the user's point of view, it looks like the water levels in the 3-D animations always change following the real test rig as shown in Fig. 11. The animation for other test rigs may be different, but they all use the similar methods to achieve the synchronization between the 3-D models and real test rigs.

Both 2-D and 3-D components are provided in NCSLab 3-D. Using 3-D animation, users can "monitor" the test rigs freely from different angles, which can increase their sense of presence. However, 2-D Controls are still necessary if users want to check the experimental process precisely. For example, the dynamical data can be plotted on a 2-D chart Control, and the instant signals can be depicted on a gauge Control.

VI. Cooling Fan Speed Control Rig: A Case Study in Wuhan University

The experimental version of NCSLab 3-D was implemented in the campus network of Wuhan University, China, in late 2011. It was applied to an undergraduate teaching module System Identification, which was optional for the final year students in the Department of Automation. There were about 90 students in the department, and 50 of them selected this module. In order to give them a chance to experience the procedures of identifying a real system, a cooling fan speed control rig shown in Fig. 12 was integrated into the NCSLab 3-D prototype. In this system, a computer cooling fan is controlled by a networked Control Unit which is connected to the NCSLab 3-D. The structure is shown in Fig. 13. The fan speed control rig can be considered as a linear system which can be represented in AutoRegressive eXogenous form as

$$\begin{aligned} y(t) + a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a) \\ = b_k u(t-k) + \cdots + b_{k+n_b} u(t-k-n_b) + e(t) \end{aligned}$$

where u is the input voltage and y is the feedback measurement from the speed sensor.

Fig. 12. Cooling fan speed control test rig.

In the remote experiment, students are required to find out the unknown parameters of the transfer function through remote experiments. In order to do so, they need to do the following jobs:

1) to design the identification algorithms by selecting proper input signal u;
2) to compile and download the algorithm to the Chinese Academy of Sciences control unit;
3) to monitor the experiments and collect both the input signal sequence u and output sequence y;
4) to analyze the collected data and find out the transfer functions by using system identification methods taught in lectures.

All the work in Steps 1)–3) can be completed in the NCSLab web-based interface. All the control algorithms in the NCSLab framework are generated by Matlab Real-Time Workshop (RTW) running in the Matlab Server. An algorithm template shown in Fig. 14 is designed to help the students customize their algorithms. In this template, a digital to analog (D/A) converter block, which gives the signal to drive the motor, and an A/D converter, which collects the speed feedback, are enclosed to establish the connection between the customized algorithms and control hardware. As shown in Fig. 14, the signal feeding into the D/A converter is marked as u, and the signal coming out of the A/D converter is named as y.

Based on the template, students can add proper input signal to design the algorithms for identification in a web-based interface. Fig. 15 shows one of the examples designed by students. These diagrams can be uploaded to the Matlab Server where they are converted into executable codes which are ready to be downloaded to the control units.

After the control unit receives the executable codes, the experiment starts automatically. Students can use the monitoring and supervising interface in the NCSLab to watch how the experiment is going on and collect necessary data. They are provided 3-D animation and various other widgets such as charts, sliders, real-time videos, etc.

As introduced in Section V-D, the motions of the 3-D models are synchronized with the test rigs through the data links. In this case, the speed of the fan is proportional to the signal y. It is noted in Fig. 14 that the signal y is enclosed in the template. All the signals in the codes generated by RTW are organized by their names. As long as the algorithms are designed to base on the given template, the signal y can always be addressed.

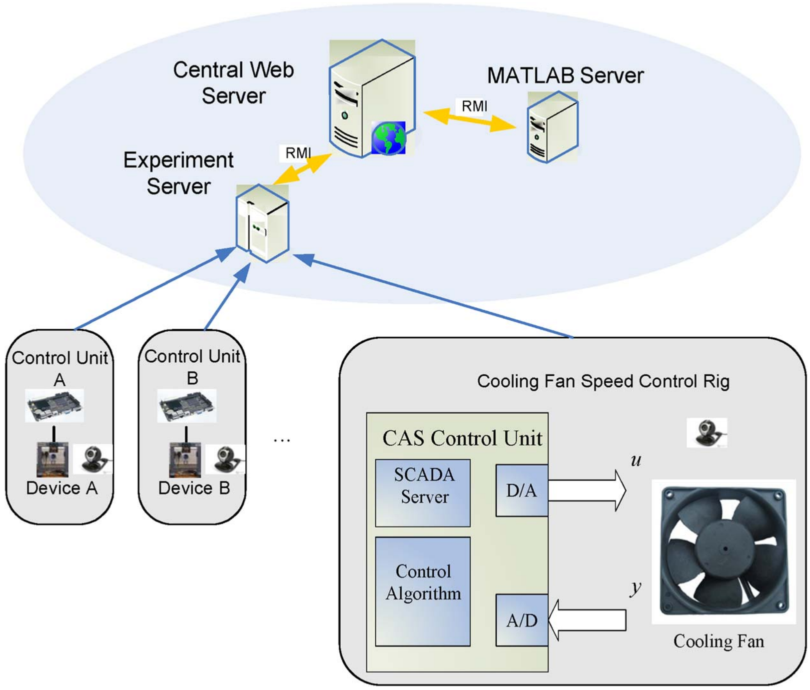


Fig. 13. Diagram of cooling fan speed rig in NCSLab 3-D.

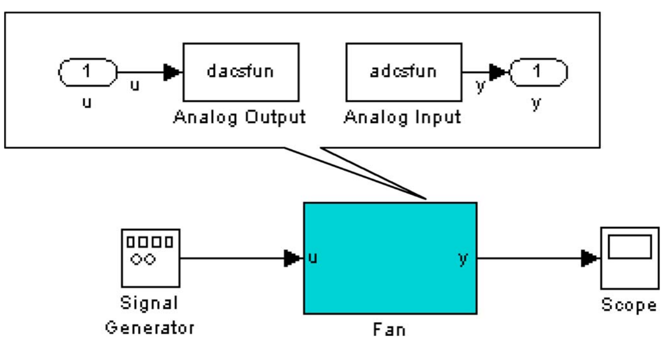


Fig. 14. Template for system identification experiments.

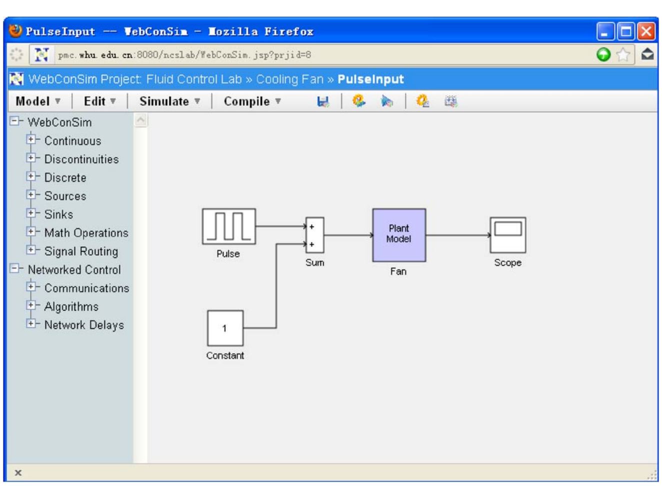


Fig. 15. Algorithm for system identification experiments.

During the experiment, the web server keeps requesting the real-time data from the control unit and transmits them to the user's side in the web browser. Based on the speed signal, the rotation of the impeller in the 3-D Flash Control can be programmed by overriding the callback function $onAnimate()$ with the following codes:

$$rotor.prependRotation(y * dt * gain, Vector3D.Z\_AXIS);$$

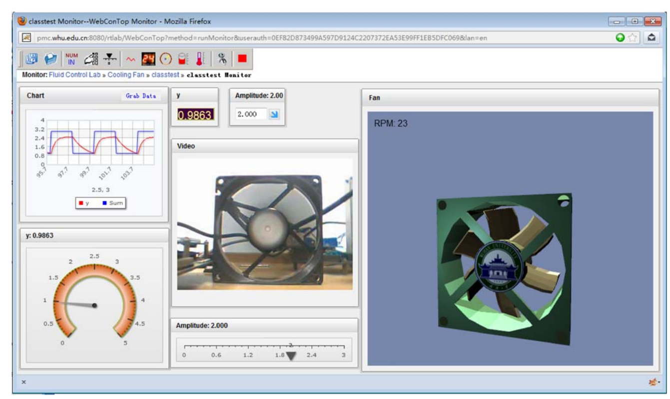


Fig. 16. Web-based monitoring interface in NCSLab 3-D.

in which y represents the speed signal, $gain$ is the coefficient between the measurement value and real fan speed, and dt is the interval between the two callbacks of function. It is clear that the multiplication of y, $gain$, and dt represents the angle that the impeller should rotate between the two neighboring callbacks.

Apart from the 3-D animation, there are many other widgets available. Users can drag and place them on the Monitoring and Supervisory Interface. Fig. 16 shows an example of a customized interface in which widgets of real-time data chart, video, slider, and 3-D animation are used.

The students who had chosen the module were required to register on the NCSLab 3-D. They were given two weeks to

TABLE I
QUESTIONNAIRE RESULT

Option	Number	Percentage
Not at all	2	6.7%
I don't know	1	3.3%
A little bit	13	43.3%
Strongly	11	36.7%
As good as hands-on experiments	3	10.0%

conduct the remote experiments and identify the mathematical model of the cooling fan. Only one test rig was allocated for them. However, all of the students had finished their homework successfully before the deadline, and the average mark was 78%. According to the students' accounts, the average time for each student to finish the laboratory work was 53 min. Five students greatly interested in it even spent more than 120 min in their homework.

When they were asked whether 3-D virtual reality can improve their sense of operating real experimental apparatuses, 90% of the students chose that it can, and around half of them gave quite positive feedback according to a questionnaire answered by 30 students. The detailed result is shown in Table I.

VII. CONCLUSION

In this paper, NCSLab 3-D, which is a web-based remote laboratory with a 3-D virtual reality interface, has been presented. The new system is based on the NCSLab framework which provides rich user interactive features and supports diversely located test rigs. In order to achieve the virtual reality interface in the web browser, 3-D Flash technology has been adopted. Using the 3-D engines, objects such as laboratory buildings, rooms, and test rigs can be embedded in users' web browsers. With the 3-D interface, users can "walk" into the sublaboratories and select test rigs. In the experiment monitoring interface, the animations of the 3-D models and real test rigs are synchronized through data links. The 3-D test rigs can be zoomed in, zoomed out, and watched from any perspectives without limitations. NCSLab 3-D has been successfully applied in the student laboratory work in the module of *System Identification* in Wuhan University. The teaching practice has shown that the use of NCSLab 3-D is able to improve both the efficiency of laboratory resources and students' sense of presence and engagement significantly.

With the rapid development of the network and virtual reality technologies, it is both challenging and critical of the NCSLab 3-D to keep up with these new progresses. The future plans for the NCSLab project have been proposed as follows.

1) *Precise synchronization between the models and real-time videos*: Video streams are from the cameras, but the real-time data are collected from the Control Unit. The networked synchronization algorithms need to be studied thoroughly to achieve this target.
2) *Integration of 3-D display technologies*: Three-dimensional television can be used to create 3-D virtual reality environment in the future.
3) *NCSLab 3-D on mobile platforms*: NCSLab 3-D can be transplanted to smart phones with 3-D acceleration capabilities.
4) *3-D interactivities*: Users are allowed to use 3-D virtual reality input devices such as data gloves to operate on the NCSLab 3-D.

REFERENCES

[1] L. Gomes and S. Bogosyan, "Current trends in remote laboratories," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4744–4756, Dec. 2009.

[2] A. K. M. Azad, M. E. Auer, and V. J. Harward, *Internet Accessible Remote Laboratories: Scalable E-Learning Tools for Engineering and Science Disciplines*. Hershey, PA: IGI Global, 2012.

[3] M. J. Callaghan, K. McCusker, J. L. Losada, J. G. Harkin, and S. Wilson, "Hybrid remote/virtual laboratories with virtual learning environment integration," in *Proc. REV Conf.*, Stockholm, Sweden, Jun./Jul. 2010.

[4] C. Maier and M. Niederstätter, "Lab2go—A repository to locate online laboratories," *Int. J. Online Eng.*, vol. 6, no. 1, pp. 12–17, 2010.

[5] T. Restivo and A. Cardoso, "Exhibition session of the exp.at'11 international conference," *Int. J. Online Eng.*, vol. 8, no. S1, pp. 4–7, 2012.

[6] J. E. Corter, S. K. Esche, C. Chassapis, J. Ma, and J. V. Nickerson, "Process and learning outcomes from remotely-operated, simulated, and hands-on student laboratories," *Comput. Educ.*, vol. 57, no. 3, pp. 2054–2067, Nov. 2011.

[7] C. Gravier, J. Fayolle, B. Bayard, M. Ates, and J. Lardon, "State of the art about remote laboratories paradigms—Foundations of ongoing mutations," *Int. J. Online Eng.*, vol. 4, no. 1, pp. 19–25, Feb. 2008.

[8] G. Sziebig, B. Takarics, and P. Korondi, "Control of an embedded system via Internet," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3324–3333, Oct. 2010.

[9] J. Sanchez, S. Dormido, R. Pastor, and F. Morilla, "A Java/MATLAB based environment for remote control system laboratories: Illustrated with an inverted pendulum," *IEEE Trans. Educ.*, vol. 47, no. 3, pp. 321–329, Aug. 2004.

[10] R. Marin, P. J. Sanz, P. Nebot, and R. Wirz, "A multimodal interface to control a robot arm via the web: A case study on remote programming," *IEEE Trans. Ind. Electron.*, vol. 52, no. 6, pp. 1506–1520, Dec. 2005.

[11] D. Lopez, R. Cedazo, F. M. Sanchez, and J. M. Sebastian, "Ciclope robot: Web-based system to remote program an embedded real-time system," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4791–4797, Dec. 2009.

[12] A. Yazidi, H. Henao, G.-A. Capolino, F. Betin, and F. Filippetti, "A web-based remote laboratory for monitoring and diagnosis of ac electrical machines," *IEEE Trans. Ind. Electron.*, vol. 58, no. 10, pp. 4950–4959, Oct. 2011.

[13] C. C. Ko, B. M. Chen, J. P. Chen, J. Zhang, and K. C. Tan, "A web-based laboratory on control of a two-degrees-of-freedom helicopter," *Int. J. Eng. Educ.*, vol. 21, no. 6, pp. 1017–1030, 2005.

[14] C. Saygin and F. Kahraman, "A web-based programmable logic controller laboratory for manufacturing engineering education," *Int. J. Adv. Manuf. Technol.*, vol. 24, no. 7/8, pp. 590–598, Oct. 2004.

[15] S. C. Wang and Y. H. Liu, "Software-reconfigurable e-learning platform for power electronics courses," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2416–2424, Jun. 2008.

[16] P. Bauer, J. Dudak, D. Maga, and V. Hajek, "Distance practical education for power electronics," *Int. J. Eng. Educ.*, vol. 23, no. 6, pp. 1210–1218, Nov. 2007.

[17] A. Valera, J. L. Diez, M. Valles, and P. Albertos, "Virtual and remote control laboratory development," *IEEE Control Syst.*, vol. 25, no. 1, pp. 35–39, Feb. 2005.

[18] A. Rojko, D. Hercog, and K. Jezernik, "Power engineering and motion control web laboratory: Design, implementation, and evaluation of mechatronics course," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3343–3354, Oct. 2010.

[19] D. Pop, D. G. Zutin, M. E. Auer, K. Henke, and H.-D. Wuttke, "An online lab to support a master program in remote engineering," in *Proc. 1st GOLC*, Rapid City, SD, 2011, pp. 1–6.

[20] S. Dormido, J. Sánchez, L. de la Torre, R. Heradio, C. Carreras, J. P. Sánchez, and M. Yuste, "Physics experiments at the UNEDLabs portal," *Int. J. Online Eng.*, vol. 8, no. S1, pp. 26–27, 2012.

[21] R. Puerto, L. M. Jiménez, and O. Reinoso, "Remote control laboratory via Internet using Matlab and Simulink," *Comput. Appl. Eng. Educ.*, vol. 18, no. 4, pp. 694–702, Dec. 2010.

[22] C. A. Ramos-Paja, J. M. R. Scarpetta, and L. Martinez-Salamero, "Integrated learning platform for Internet-based control-engineering education," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3284–3296, Oct. 2010.

[23] G. Farias, R. De Keyser, S. Dormido, and F. Esquembre, "Developing networked control labs: A Matlab and easy Java simulations approach," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3266–3275, Oct. 2010.

[24] J. Garcia-Zubia, P. Orduna, D. Lopez-de-Ipina, and G. R. Alves, "Addressing software impact in the design of remote labs," *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4757–4767, Dec. 2009.

[25] V. J. Harward, J. A. Del Alamo, S. R. Lerman, P. H. Bailey, J. Carpenter, K. DeLong, C. Felknor, J. Hardison, B. Harrison, I. Jabbour, P. D. Long, T. Mao, L. Naamani, J. Northridge, M. Schulz, D. Talavera, C. Varadharajan, S. Wang, K. Yehia, R. Zbib, and D. Zych, "The iLab shared architecture: A web services infrastructure to build communities of Internet accessible laboratories," *Proc. IEEE*, vol. 96, no. 6, pp. 931–950, Jun. 2008.

[26] A. Leva and F. Donida, "Multifunctional remote laboratory for education in automatic control: The CrAutoLab experience," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2376–2385, Jun. 2008.

[27] J. Fayolle, C. Gravier, N. Yankelovich, and E. Kim, "Remote lab in virtual world for remote control of industrial processes," in *Proc. IEEE ICME*, Barcelona, Spain, Jul. 11–15, 2011, pp. 1–4.

[28] L. Gomes and J. García-Zubía, *Advances in Remote Laboratory and e-Learning Experiences*. Bilbao, Spain: Univ. Deusto, 2007.

[29] A. G. Vicente, I. B. Muñoz, J. L. L. Galilea, and P. A. R. del Toro, "Remote automation laboratory using a cluster of virtual machines," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3276–3283, Oct. 2010.

[30] M. Casini, D. Prattichizzo, and A. Vicino, "The automatic control telelab: A web-based technology for distance learning," *IEEE Control Syst. Mag.*, vol. 24, no. 3, pp. 36–44, Jun. 2004.

[31] M. J. Callaghan, J. Harkin, T. M. McGinnity, and L. P. Maguire, "Intelligent user support in autonomous remote experimentation environments," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2355–2367, Jun. 2008.

[32] D. Gillet, A. V. N. Ngoc, and Y. Rekik, "Collaborative web-based experimentation in flexible engineering education," *IEEE Trans. Educ.*, vol. 48, no. 4, pp. 696–704, Nov. 2005.

[33] W. Hu, G. P. Liu, D. Rees, and Y. Qiao, "Design of web-based real-time control laboratory for diversely located test rigs," in *Proc. IFAC World Congr.*, Seoul, South Korea, 2008, pp. 12 643–12 648.

[34] W. Hu, G. P. Liu, D. Rees, and Y. Qiao, "Design and implementation of web-based control laboratory for test rigs in geographically diverse locations," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2343–2354, Jun. 2008.

[35] Y. Qiao, G. P. Liu, G. Zheng, and W. Hu, "NCSLab: A web-based global-scale control laboratory with rich interactive features," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3253–3265, Oct. 2010.

[36] W. Huang, R. Li, C. Maple, H.-J. Yang, D. Foskett, and V. Cleaver, "A novel lifecycle model for web-based application development in small and medium enterprises," *Int. J. Autom. Comput.*, vol. 7, no. 3, pp. 389–398, Aug. 2010.

[37] J. E. Corter, J. V. Nickerson, S. K. Esche, C. Chassapis, S. Im, and J. Ma, "Constructing reality: A study of remote, hands-on, and simulated laboratories," *ACM Trans. Comput.-Hum. Inter.*, vol. 14, no. 2, p. 7, Aug. 2007.

[38] C. Carter, A. E. Rhalibi, and M. Merabti, "Development and deployment of cross-platform 3D web-based games," in *Proc. DESE*, London, U.K., 2010, pp. 149–154.

[39] F. Lamberti, A. Sanna, and E. A. H. Ramirez, "Web-based 3D visualization for intelligent street lighting," in *Proc. 16th Int. Conf. 3D Web*, New York, 2011, pp. 151–154.

[40] M. Over, A. Schilling, S. Neubauer, and A. Zipf, "Generating web-based 3D city models from open street map: The current situation in Germany," *Comput., Environ. Urban Syst.*, vol. 34, no. 6, pp. 496–507, Nov. 2010.

[41] F. Marino, P. De Ruvo, G. De Ruvo, M. Nitti, and E. Stella, "HiPER 3D: An omnidirectional sensor for high precision environmental 3D reconstruction," *IEEE Trans. Ind. Electron.*, vol. 59, no. 1, pp. 579–591, Jan. 2012.

[42] W. Hu, H. Zhou, and Q. Deng, "Design of web-based 3D control laboratory," in *Proc. 2nd Int. Conf. Intell. Control Inf. Process.*, Harbin, China, 2011, pp. 590–594.

[43] C. C. Ko and C. D. Chang, *Interactive Web-Based Virtual Reality With Java 3D*. Hershey, PA: IGI Global, 2008.

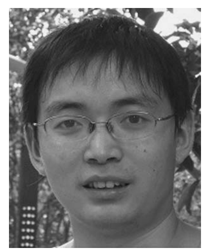
[44] J. García-Zubía, J. Irurzun, P. Orduña, I. Angulo, U. Hernández, J. Ruiz, M. Castro, and E. San-Cristóbal, "Second Lab: A remote laboratory under Second Life," *Int. J. Online Eng.*, vol. 6, no. 4, pp. 34–39, Feb. 2010.

[45] E. S. Aziz, S. K. Esche, and C. Chassapis, "An interactive game-based engineering laboratory," *World Trans. Eng. Technol. Educ.*, vol. 8, no. 2, pp. 131–136, 2010.

[46] M. J. Callaghan, K. McCusker, J. Losada, J. Harkin, S. Wilson, J. Dugas, S. Demots, F. Desbois, A. Fouquet, and F. Sauviat, "Game-based strategy to teaching electronic and electrical engineering in virtual worlds," in *Proc. IEEE 2nd Int. Consum Electron. Soc. Games Innov. Conf.*, 2010, pp. 1–10.

[47] R. Marcelino, J. B. Silva, A. V. Fidalgo, L. Schaeffer, and J. B. M. Alves, "Virtual 3D worlds and remote experimentation: A methodology proposal applied to engineering students," in *Using Remote Labs in Education*. Bilbao, Spain: Univ. Deusto, 2012, pp. 349–374.

[48] I. Arsov, "A framework for distributed 3D graphics applications based on compression and streaming," Ph.D dissertation, Telecom SudParis, Evry, France, 2009.



Wenshan Hu received the B.S and M.Sc. degrees in control theory and applications from Wuhan University, Wuhan, China, in 2002 and 2004, respectively, and the Ph.D. degree in control engineering from the University of Glamorgan, Pontypridd, U.K., in 2008.

He was a Visiting Research Fellow at the Faculty of Advanced Technology, University of Glamorgan, during 2008 to 2010. In June 2010, he joined the Department of Automation, Wuhan University, as an Associate Professor. His main research interests include networked control systems and network-based control laboratories.

Guo-Ping Liu (M'97–SM'99–F'11) received the B.Eng. and M.Eng degrees in automation from the Central South University of Technology, Changsha, China, in 1982 and 1985, respectively, and the Ph.D. degree in control engineering from University of Manchester Institute of Science and Technology, Manchester, U.K., in 1992.

He is the Chair of Control Engineering in the University of Glamorgan, Pontypridd, U.K. He has been a Professor in the University of Glamorgan since 2004, a Hundred-Talent Program Visiting Professor of the Chinese Academy of Sciences, Beijing, China, since 2001, and a Changjiang Scholar Visiting Professor of the Harbin Institute of Technology, Harbin, China, since 2008. He is the Editor-in-Chief of the International Journal of Automation and Computing. He has more than 400 publications on control systems and authored/coauthored seven books. His main research areas include networked control systems, nonlinear system identification and control, advanced control of industrial systems, and multiobjective optimization and control.

Prof. Liu is a Fellow of Institution of Engineering and Technology.

Hong Zhou received the B.S. degree from the Central South University of Technology (now the Central South University), Changsha, China, in 1982, the M.Sc. degree from Chongqing University, Chongqing, China, in 1988, and the Ph.D. degree from Wuhan University, Wuhan, China, in 2006.

He was an Assistant Engineer with Changshou Power Plant, Chongqing, during 1982 to 1985. He has been with the Wuhan University of Hydraulic and Electrical Engineering (now a part of Wuhan University), Wuhan, since 1992. He was the Head of the Department of Automation, Wuhan University, during 2002 to 2006 and has been a Professor in the department since 2000. He has worked on more than 30 academic research and industrial technology projects. His main research interests are smart grid and networked control systems.