Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Remote Labs and Resource Sharing in Control Systems Education ⋆

Pavel Burget * **Ondřej Fiala** ** **Tomáš Fencl** *** **Lukáš Moc** ****

* *Department of Control Engineering, Faculty of Electrical Engineering, Czech Technical University, Prague, Czech Republic (burgetpa@fel.cvut.cz)*
** *(fialao@fel.cvut.cz)*
*** *(fenclt@fel.cvut.cz)*
**** *(lukas@lukasmoc.cz)*

Abstract: The goal of this paper is to present new extensions to our remote laboratory, which has been used for more than 4 years in undergraduate courses in the Department of Control Engineering. The remote lab serves as a means of a distant-programming environment, which provides users the possibility to use control systems, widely employed in industrial applications. We have reworked the remote lab completely, in line with our long-term experience from previous lab utilisation, while keeping in mind the current trends in mobile computing. Our unique combination of the access-control system, flexible management of the laboratory infrastructure and virtual remote desktops allows users to work absolutely independently of the place and time. The students can use simulation software (such as a programmable logic controller simulator), as well as remotely control the physical experiments placed in the lab. As a result of our work, the users of the lab gain a complex, while still simple-to-use, means to learn and practise on experiments, which would otherwise be inaccessible to them.

Keywords: Control education; Laboratory; Resource allocation; Distant programming.

1. INTRODUCTION

There has been a remote lab in the Department of Control Engineering (DCE) at the Czech Technical University in Prague (CTU) for more than four years, being used by students of undergraduate courses. In Burget et al. [2004], where the remote lab is described, some topics are mentioned that discuss the differences between the virtual and remote labs, and then more details of the reservation and access system are given. After several years of experience, we have completely reworked the lab as the previous solution was not fulfilling the user demands any more.

In this paper, we present this new solution that has been built on the idea of remote labs to control the physical processes, and to apply commercial controllers used in industrial applications. Up to now, these prerequisites, which the previous remote lab was based on, seem to be correct for students to get as close as possible to the industrial-application means. What has changed is the way how students/users get access to the resources in the lab and how the security and exclusivity of the resource allocation are ensured. This unique approach of ours is focused on in this submission.

The organisation of the paper is as follows: In Section 2 we present the background of the remote lab and some related work to form the framework of our approach. Section 3 discusses other issues of remote labs, mainly in comparison with virtual labs. We shortly describe one of the models and summarise our experience with the previous remote lab in the DCE, its advantages and disadvantages. Section 4 forms the core of this paper, giving details about the new concept of the lab access-control and resource-allocation system. In Section 5, we conclude the paper and outline our expectations and the perspective of future developments.

2. BACKGROUND AND RELATED WORK

There have been attempts to perform remote operation of physical models, robots or other kinds of controllers for the purpose of tele-operation, education, etc. To look back in history a little, Goldberg et al. [1995] describe their approach to use the Web in its relatively early stages to operate a manipulating arm remotely. Later on, there have been many papers and articles that presented various ways how to use the Internet for remote access to physical or virtual models. [1] Ewald and Page [2000] describe their approach to remote control over the Internet. It is based on a client-server model, where the server is connected to the

⋆ The whole work was supported by the Czech Ministry of Education, Youth, and Sports under the EuSophos project No.2C06010, and further supported by the European Social Fund and co-financed by the state budget of the Czech Republic under project CEPOT - CZ.04.3.07/4.2.01.1/0045.

[1] Of course, there have also been many ways published, on how to present educational materials to the students in various forms of e-learning courses. However, this area is not focused on in this paper, although it is a crucial part of distant education.

controlled laboratory experiments over an instrumentation bus. The server presents so called virtual instruments to the remote user that are in fact a graphical representation of physical devices. Casini et al. [2001] in their paper present a remote control engineering laboratory whose aim is to offer remote users the ability to design controllers and test them on a physical plant such as a DC motor for position control or tank equipment for level control. The Matlab/Simulink environment is used for the controller design and Real-time Workshop is used to apply the designed control law to the physical plant. The remote user communicates with the Matlab environment through a Java-applet interface over a TCP/IP connection. A similar setting is described by Guimarães et al. [2002] and Jordan et al. [2002]. Their remote lab facilities contain dedicated computers, each of them is connected to a physical plant, running a pre-designed controller. A Java-based client interface allows users to set the controller parameters. Garbus et al. [2006] present a facility similar to the one described in Ewald and Page [2000], just a customer-specific module for process instrumentation is used. LabView or Matlab environments can be used here to provide the client interface.

A little bit aside, but definitely worth mentioning, is the work of Goldberg et al. [2003], solving the problem of collaborative cooperation of several remote users. Networked spatial dynamic voting is used to generate a final command for the manipulator that is represented by a manipulating robot arm or by a human called a tele-actor.

Finally, for this part of Section 2, we shortly summarise the results from Burget et al. [2004]. The remote lab described there, is based on a programmable logic controller connected to a physical plant. Two physical plants were regarded – a pneumatic and an electromagnetic colour ball sorter. A user management system was described (see Fig. 1) that offered the possibility to make a reservation for a certain time slot to be dedicated solely and exclusively to the individual user, having made that reservation. Such a combination of real-time control, remote access and time slot reservation system with user authentification

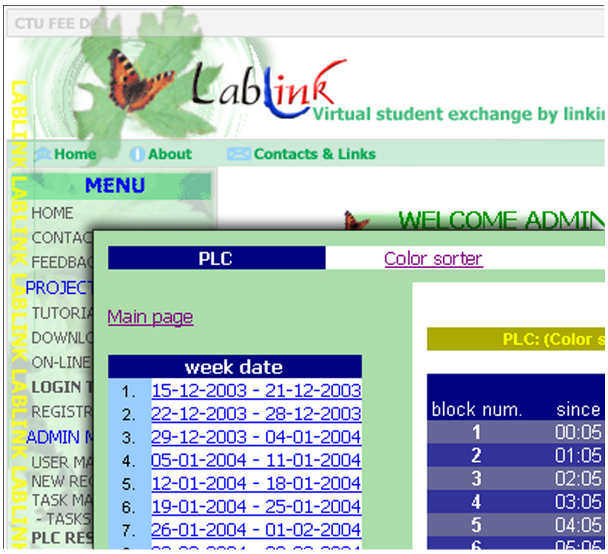

Fig. 1. The old LabLink portal

are principal requirements for a successful utilisation of a remote lab. However, in many publications and remote-lab applications, the reservation and authorisation system description is often omitted.

This is not the case with our method. Already in the previous project of the remote lab, and even more in the project presented here, we have put in a great deal of attention to this part and thus, have created a system that is unique in its abilities to allow users utilising the remote lab resources. We have also aimed our solution towards the security features to disable potential hackers to attack inner network and computer structures. This approach is described in more details in Section 4.

3. FACTS ABOUT REMOTE LABS

As stated before, remote labs provide users an opportunity to use the equipment of the lab, i. e. controllers, physical experiments, etc., without being present in the lab itself. The utilisation of the models is much higher because even users from other institutions can use the remote-lab resources for their laboratory practice. However, there are some requirements[2] on the models that must be counted on during the model design.

There are also certain requirements on the client-side equipment and client-server relations, and to pinpoint the advantages of our approach described later on, we shall discuss these features and relations a little bit here.

- It is unsuitable, if the user has to install a non-standard software on his/her computer. An acceptable solution is to use a web browser or a Java-based application because it can run on almost any computer connected to the Internet. They are also operating system (OS) independent, i. e. users running Windows, Linux, MacOS, etc. are able to exploit the possibilities of the remote lab. However, most industrial programming environments are installed as standalone applications.
- In most cases, the real-time control over the Internet standard protocols[3], i. e. the control software would run on the client side, cannot be performed because of the nature of the IP protocol suite. There are indeed some exceptions such as a video stream transfer using UDP (User Datagram Protocol) under certain QoS (Quality of Service) guarantees, but in many applications with strict real-time requirements another solution must be chosen. Such a solution relies on the fact there is a dedicated computer in the lab that performs the control of the physical facility and just accepts commands from the remote user. The command transfer of this kind is not time-critical.
- An important issue to care about is the availability of the industry-based equipment. Such equipment, if available, can increase the usability value of the lab because the students get better training for possible future industrial applications. The other solution, which represents a specialised one-purpose piece of hardware developed solely for the application of the

[2] These requirements are described in Burget et al. [2004] extensively and we are not dealing with them in this paper.
[3] Real-time Ethernet protocols such as PROFINET IO or Ethernet Powerlink are not taken into account in this paper.

remote lab, may offer some better customisation opportunities on one hand, however, they may contradict the intention to get as close as possible to real-life applications on the other.

Another important issue of remote laboratories is the fact whether we use physical or virtual models, i. e. if we call the lab remote or virtual, as described e. g. in Casini et al. [2003]. In Burget et al. [2004] we definitely preferred remote labs. However, after some years of experience with the remote lab, we have assumed virtual models to be a suitable supplement of the remote labs. From our point of view, the virtual models are especially useful if they represent a simplified version of a facility that is in the remote lab which the users can control. Therefore, users can prepare their control algorithm and after testing it they can pass it over to work with the physical model. In such a way, they spend less time with the physical model and thus:

- They can debug most errors, mainly simple logical ones, on the virtual model. It is even possible to run some automatic testing and verification algorithms (see e. g. Šusta and Burget [2008]) in order to get a more reliable control application.
- The utilisation of physical models is more effective because the users do not have to spend all the debugging time there, as explained above.
- More users can work concurrently with the virtual models because they are not limited by the available physical resources, as in the other case.

3.1 Experience to Date

The students in DCE usually like working in the remote lab due to the several advantages such work involves. On one hand, there is the possibility to use industry-like equipment and on the other the equipment can be accessed from almost everywhere. The Internet transmission speed is not critical in this case because the real-time control is executed locally, i. e. by the local logic controller or control computer. Another aspect of the advantage is the fact the time the model can be accessed is not limited.

The up-to-date system of the remote models has been used extensively in the education of the students in the department. Over the years, several aspects of remote-lab usage have occurred which have caused us to change the lab. These aspects are as follows:

- The way of programming, i. e. the way of using the programming environment, was not the same for all the models. Some models required installing the programming environment on the client computer while others used the remote desktop connection. However, the remote computer, to which the remote connections were made, did not possess any server administration tools and thus the management of the system was complicated.
- The authentication was based on IP address detection and IP forwarding rules. This method worked correctly if the client computer was not behind a proxy or a part of a network with address translation. In such a case several computers can appear as having the same IP address and thus the access of one computer can be interrupted by another one on the same network.
- The users who reserve multiple time slots and do not use them afterwards, were not penalised. The possibility arises that the model does not get used, although the current time slot was reserved due to the user, who made the reservation, failing to log in.
- Last, but not least, the mechanical construction of the pneumatic colour ball sorter (see Burget et al. [2004]) needed some improvements.

In the following section, which is the core of this paper, we present the new concept of the remote lab together with a sketch of the newly designed construction of the pneumatic colour ball sorter.

4. NEW CONCEPT OF THE REMOTE LABS

The new concept lies in the fact that a central server exists, which not only controls the user management and authentification but also performs the virtual remote desktop creation upon the user's demand. Such a solution offers full administration capabilities in one place, including software installation, the possibility to repair a corrupted installation, send messages to users that are logged-in, etc. The principles of this new concept comprise the network infrastructure and the server application.

4.1 Network Infrastructure

The idea of the network infrastructure is depicted in Fig. 2. The remote lab consists of a group of physical models, each model controlled by a programmable logic controller (PLC). Such models are connected to a central managed switch and, through this switch, to the main server of the remote lab (LabLink server). The users with physical models utilise virtual machines, which run on the LabLink server. Additionally, there is also an application server (see Section 4.2).

The pair server-switch thus forms the remote lab's core that is fully self-supporting, providing all the necessary means for the remote users to get exclusive access to a selected model. Such a settlement allows it to be simply applied and can be reused without having to perform any additional complicated configuration inside the system. Therefore, the system is repeatable and can thus be easily migrated to other labs. The individual labs could also be connected to each other. Such an interconnection would allow users to utilise models from different sites that communicate with each other. The management of the system can be centralised or distributed among the sites of the system, but in both cases it would ensure a unified access to all the interconnected lab resources.

Internal, as well as external users connect to the system uniformly via the LabLink server. The connection is built above the usual TCP/IP communication using common protocols such as HTTP, HTTPS and RDP (Remote Desktop Protocol). To access a physical model a user logs in using the RDP to one of the virtual machines. This virtual machine is started at the server, exclusively for this user, having been configured according to the user requirements, and containing the user's personal data, having been copied to the configuration. At the same time,

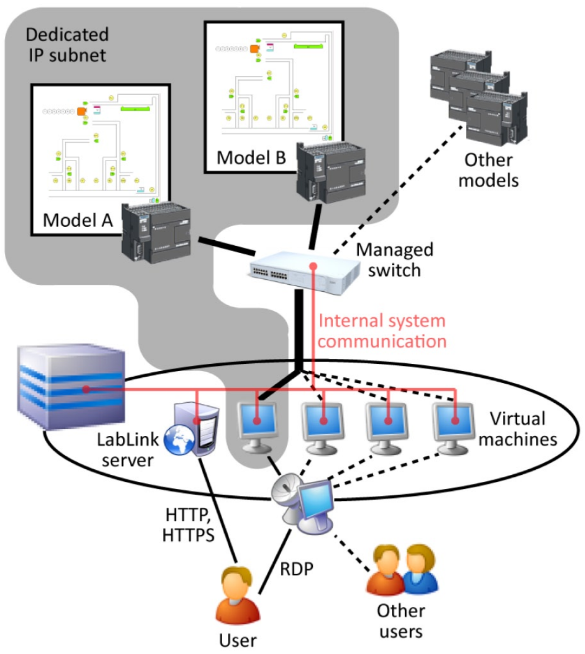

Fig. 2. Virtual network

a dedicated IP subnet is created that contains the virtual machine, and the physical models, which were selected by the user during reservation. The user can access and program the models using a corresponding programming environment such as Simatic Step7, WAGO CoDeSys, etc., which have been installed on the virtual machine. In fact, they were installed in the image, which the virtual machine is dynamically created from.

The situation where a user connects to the server using the RDP, is shown in Fig. 2. If the connection is accepted by the server, a virtual network is dynamically created for this user. The virtual network contains two models (model A and model B) and the virtual machine, started especially for this user. The figure also shows the internal communication among the LabLink server, individual virtual machines and the managed switch.

All the models in the remote lab are operated using a web service, running on the LabLink server. The web service keeps various tools at its disposal, allowing one to handle all types of the PLCs connected to the individual models. A specific set of tools is created for every PLC type present in the lab, that offer functions such as determining the IP address of a selected PLC, downloading a project into the PLC, and controlling the model over a dedicated communication protocol. Such tool sets are necessary to ensure the functionality of the lab, meaning resetting a model after a user's reservation period runs out, and running a model in the demo mode. Resetting a model means actually bringing the model in a defined initial state, and resetting the connected PLC. The demo mode of a model is used to demonstrate the capabilities of the model on one hand, and to test all the functions of the model manually on the other hand. It is also possible to run an exemplary PLC application, controlling the model.

4.2 Server application

The application part of the system is formed by a web application, providing access to the main functions of the remote lab. The application is implemented in Ruby and in the Ruby On Rails web-based application framework, and its data is stored in a SQL database. The application part of the system runs on the LabLink server (see Fig. 2).

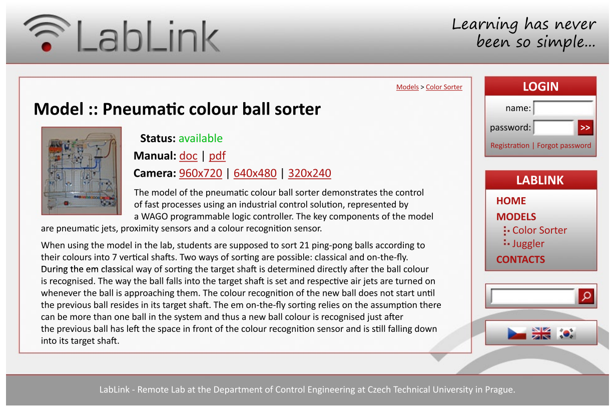

Fig. 3. The new LabLink portal

The application takes care of the user management and registration, the management of the remote lab resources, the reservation of selected resources for a specific user at a specific time, the on-line communication with the users currently connected, and provides an overall review of the current state of the remote lab.

The main function of the application is the reservation of the remote-lab resources. The reservation is accessible only to registered users and is performed in three steps:

(1) The user selects one or more physical model, which he/she wants to work with. Only models, which the user is entitled to work with, are offered.
(2) The user chooses a configuration of the virtual machine. In preference, the system offers such configurations that are suitable and/or optimal for the current user selection of the physical models. It is worth mentioning here that the user can select not only several physical models but also several virtual machines. In such a configuration, the intercommunication among the models and virtual machines can be programmed and tested too.
(3) The user selects a specific time when the chosen combination of the lab resources (models and virtual machines) can be reserved for this user. This selection is performed in a calendar, which the user can see and which highlights the available time slots, i. e. the time slots in which the chosen resources are available.

Besides the standard reservation of the lab resources by a user, described above, the system can block specific resources by a superior user (a teacher in the lab) to be used during ordinary education. Such a blockage is performed before the semester starts. In case, the teacher finds out during the semester, he/she needs to change the current blockage, he/she can cancel the reservation of other users (students) for such models. For example, it could be the case when such models produce unpleasant or too loud a noise and distort the education in the lab.

Another function of the application part of the system is the on-line communication with running virtual machines and e-mail communication with the users of individual physical models. The e-mail communication with the users allows the lab administrator to tell the current users that a certain model is out of order or is being repaired. Thus, the users, who have reserved the model, can move their reservation to another available time. The on-line communication is used to notify users that the end of their reserved time slot is approaching and that they should finish their work. The administrator or a teacher can also ask a specific user in this way too, for example, to finish his/her work. If the user does not obey this request, the administrator or the teacher can disconnect the user from his/her model and/or virtual machine.

4.3 User Management

Ordinary (registered) and also anonymous [4] users are distinguished by the system. The advantage of the anonymous users is they can test the systems, e. g. a selected virtual model, without the necessity to register and wait for the registration confirmation. However, such users can reserve no more resources than 1 model with 1 desktop, and no longer than 24 hours ahead. Still, there is the advantage of using the remote lab exactly at the time an anonymous user connects, if there is a free slot. On the contrary, the registered users are limited neither by the number and type of resources (virtual/physical) nor by the time period.

A penalisation system has been introduced as well. It allows for handicapping users that "do not behave" correctly. A user is penalised whenever he/she does not log in the system even if a time slot has been reserved for him/her. In such a case, all the user's future reservations are cancelled automatically and for the next 48 hours the user is not allowed to reserve any resources of the lab. However, such a user can watch the reservation calendar in a read-only mode or work as an anonymous user.

Fig. 4 shows typical front-end use cases, which formally describe how the LabLink portal is used. In addition to the registered user described above, there are users of type *teacher* and *administrator* in Fig. 4. The teacher can manage user reservations, send messages to the users and set the status of individual resources besides of being able to inherit the abilities of the *registered* user. On the top of it, the administrator can manage the resources of the lab. There is one type of user more – the *unidentified* user. This is a casual user, who encounters the LabLink information portal, while browsing the Web, and wants to get more information about it.

4.4 Model reconstruction

The model of the pneumatic colour ball sorter demonstrates the control of fast processes using an industrial control solution, represented by a WAGO programmable

[4] The anonymous users are not completely anonymous because they must give their e-mail address, which they use to log into the system.

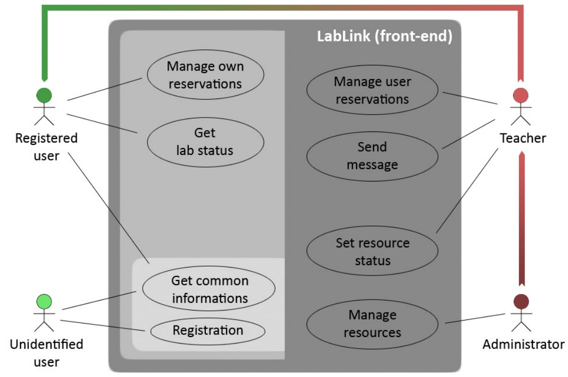

Fig. 4. Selected front-end use cases

logic controller. The key components of the model are pneumatic jets, proximity sensors and a colour recognition sensor. When using the model in the lab, students are supposed to sort 21 ping-pong balls according to their colours into 7 vertical shafts. Two ways of sorting are possible: classical and on-the-fly. During the *classical* way of sorting the target shaft is determined directly after the ball colour is recognised. The way the ball falls into the target shaft is set and respective air jets are turned on whenever the ball is approaching them. The colour recognition of the new ball does not start until the previous ball resides in its target shaft. *On-the-fly* sorting relies on the assumption there can be more than one ball in the system and thus a new ball colour can be recognised just after the previous ball has left the space in front of the colour recognition sensor and is still falling down into its target shaft. This type of sorting is more difficult but also more interesting. The increased difficulty lies in the fact that each ball in the system must "remember" its path during the whole time it's falling down into the target shaft.

The above description relates to the newly redesigned model, which is supposed to replace the old one, described e.g. in Burget et al. [2004]. The reasons for the reconstruction were especially due to the following:

- The colour recognition sensor was not robust enough to deal with smudged balls in order to recognise their colour correctly.
- Air swirls occurred in the shafts in certain cases and the control program had to deal with such situations.

Due to the above mentioned disadvantages, the sorting process produced incorrect sorting and had to be slowed down. Thus, a new model was designed, which has corrected the imperfections and which allows on-the-fly sorting. The structure of the model can be seen in Fig. 5 First of all, we have installed a new colour recognition sensor, which is able to recognise the ball colour within 30 ms after the ball stabilises in front of the sensor. We have added proximity sensors in every shaft to be able to detect correct/incorrect sorting more easily. Contrary of the previous model, the air jets are used to move the balls into the upper container (air jet J_8), and to mix them there (air jets J_7). One of the J_7 air jets is also used to move the balls in the lower container to get just one ball in front of J_8. Difficulties may arise as long as there is more than one ball in front

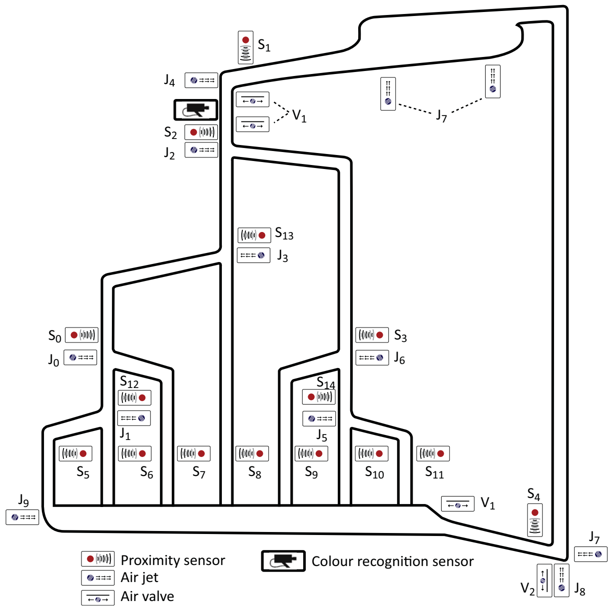

Fig. 5. The redesigned pneumatic colour ball sorter

of J_8. To avoid such a situation, air valve V_2 has been put at the end of the lower container just before a ball gets in front of air jet J_8. Alternatively, the students may be encouraged to find such a frequency of the blow strokes of J_8 to gradually increase the height the balls reach, in order for them to finally end up in the upper container.

4.5 Remote Lab Experience

Up to now, 20 students have worked with the new model. Its behaviour seems to be much better than in the case of the old one. Most of the students also decided to perform on-the-fly sorting, which is more difficult but also more interesting and impressive. However, the biggest contribution is the new concept of the remote lab itself. The possibility to combine the physical and virtual models in one environment is unique and opens further possibilities in the model development and utilisation. The concept has proved to be correct and after minor improvements during the lab's two-month operation it is ready for uninterrupted activity.

5. CONCLUSION

We have created a new concept of the remote lab which offers a unified environment to all users (internal and external) independently of the type of model and kind of programming environment. We expect such a remote lab system will supersede the existing one and will offer the users much more comfort and reliability.

5.1 Future Work

Subsequently, we are going to improve the server application to be more robust and to offer users and the administrator staff more comfort. The intended enhancements include, for example, automatic virtual model administration, integrated user communication, etc. Other physical models are to be constructed as well, such as the one of a small brewery. This model is to be the most important highlight of our lab in the future because it is going to integrate the design of control laws, industrial control and visualisation in one remote lab. The steps, performed up to now, are essential for such future developments.

An interesting area is the integration of real-time Ethernet protocols in the remote lab. Such protocols usually allow one to combine real-time data transfer and non-real-time standard IP protocols. It is also possible to include a mobile device such as a mobile phone, PDA or a mobile operator panel to operate the models in the lab for local demonstrations, while still using the means that the remote lab offers.

Another direction of further development lies in spatial expansion, i. e. creating a distributed remote lab. This concept is outlined shortly in Section 4.1 and allows different laboratories to be connected into one distributed laboratory. In such a way, different institutes can agree on a way of cooperation during which various complex models can be constructed and after incorporating them into the remote lab, shared among users not only from the concerned institutes but literally from all over the world.

REFERENCES

P. Burget, O. Dolejš, Z. Hanzálek, and B. Kirchmann. Remote programming of control systems. In *2nd IFAC Workshop on Internet Based Control Education*, 2004.

M. Casini, D. Prattichizzo, and A. Vicino. The automatic control telelab: a remote control engineering laboratory. In *40th IEEE Conference on Decision and Control*, 2001.

M. Casini, D. Prattichizzo, and A. Vicino. The automatic control telelab: A user-friendly interface for distance learning. *IEEE TRANSACTIONS ON EDUCATION*, 46(2):252–257, 2003.

H. Ewald and G. F. Page. Performing experiments by remote control using the internet. *Global Journal of Engineering Education*, 4(3), 2000.

R. U. Garbus, I. J. O. Aguirre, R. C. Sánchez, and O. R. Pureco. Virtual remote lab for control practice. pages 361–366, 2006. doi: http://doi.ieeecomputersociety.org/10.1109/CERMA.2006.125.

K. Goldberg, M. Mascha, S. Gentner, N. Rothenberg, C. Sutter, and J. Wiegley. Desktop tele-operation via the world wide web, 1995. URL `citeseer.ist.psu.edu/goldberg95desktop.html`.

K. Goldberg, D. Song, and A. Levandowski. Collaborative teleoperation using networked spatial dynamic voting, 2003. URL `citeseer.ist.psu.edu/goldberg03collaborative.html`.

B. Guimarães, A. S. Souza, H. L. Gosmann, and Adolfo Bauchspiess. Internet based remote laboratory: The level control of three coupled water reservoirs. IFAC, 2002.

R. Jordan, A. Martinez, E. Tamura, M. Martinez, and C. Olarte. A remote laboratory for an automatic control electromechanical system. International Conference on Engineering Education, 2002. URL `citeseer.ist.psu.edu/jordan02remote.html`.

R. Šusta and P. Burget. Testing of control programs in distance education. In *17th IFAC World Congress*, 2008.