Virtual Robotic Laboratory: Applying web-based teleoperation technologies to distance training in robot manipulator programming

C.S. Tzafestas[1], *M. Alifragis*[1], *N. Palaiologou*[2], *S.C.A. Thomopoulos*[3],
A.-E. Exarchou[4], *A. Kroys*[5] *and R. Kunicke*[5]

[1] National Technical University of Athens (NTUA), School of Electrical and Computer Engineering
[2] University of Piraeus, Department of Technology Education and Digital Systems
[3] National Center for Scientific Research 'Demokritos', Instit. of Informatics and Telecommunications
[4] IS Integrated Product Information (K-DOE-5), VW Group, Wolfsburg
[5] Otto-von-Guericke University Magdeburg, Faculty of Computer Science

Key words: *Virtual Laboratory, Web-based Telerobotics, Robot Manipulation Teleprogramming, Distance Training, Remote Experimentation*

Abstract:

This paper describes the development and experimental evaluation of a virtual and remote laboratory platform aiming at the distance training of students in robot manipulator programming. The graphical user interface is based on Java technologies and incorporates, among other control panels, a "virtual robot panel", providing 3D visualization of both the commanded (preview animation) and the current robot configuration supporting both direct teleoperation and indirect teleprogramming as remote robot control modes. The user interface also incorporates a "Virtual Pendant" panel, providing an exact emulation of the robot's Teach Pendant functionality. A pilot experimental study was conducted to assess system performance, in terms of remotely training students to program robot manipulation tasks with the Teach Pendant. The robot used in the experiments was a SCARA-type AdeptOne manipulator with 4 degrees of freedom. Analysis of the results obtained from this pilot study is encouraging, showing that this virtual laboratory concept can indeed be applied quite efficiently for training students remotely.

1 Introduction

During the last decade, many "distance-learning" platforms and applications have been developed, to enable teaching from a remote location in a synchronous or asynchronous e-learning mode. The development of such applications is often based on some type of tele-conferencing (video/audio streaming) platform, with an MCU (multi-point conferencing unit) at the core of the system, enhanced by many software features, such as application sharing or other functionalities forming "virtual classroom" web-spaces. We can, thus, say that nowadays attending and participating in classroom lectures or seminars remotely is a technologically feasible goal, as related technologies are mature enough and many application platforms have already been established as a standard.

However, in many cases, exchanging audio/video streams, sharing educational material (such as presentation slides) in a synchronous or asynchronous way, or interacting in a "virtual classroom" space are often not adequate to complete an efficient educational program. A typical example is teaching in engineering disciplines, where hands-on laboratory experimentation is still considered to be irreplaceable and absolutely necessary for enhancing and completing classroom lectures. In this context, a number of "*virtual laboratory*" projects have been initiated during the last 4-5 years, on a national or international basis, aiming to teach fundamental concepts in different engineering fields through the remote operation and control of specific experimental facilities. A typical example is the project ReLAX (remote laboratory experimentation trial), funded by the European Commission within the IST framework, which studied the feasibility of making remote experimentation available as a component in distance learning, both from a technological point of view as well as from an economic perspective [1]. This project proceeded with the evaluation of a new business model, the so-called "experiment service provider model", proposing the establishment of a global remote laboratory network (the CyberLab network [2]). A continuation of this effort is the eMersion project aiming to study the deployment of innovative pedagogical scenarios and flexible learning resources for carrying out virtual or remote experiments via Internet [3]. Similar activities towards the development of virtual and remote laboratory systems are also carried out by many other academic institutions, covering various engineering fields, ranging from electronics [4] and control [5], to a larger variety of mechanical and chemical engineering experimental set-ups [6].

Experience acquired from this work and from other similar initiatives reveals the difficulties and the challenges associated with the introduction and deployment of distance laboratory modules. From a technical point of view, such a goal requires adaptation of existing equipment, which must often be performed on a "task-specific" way, and involves interfacing through the network of many different physical devices and diverse experimental equipment needed to complete a real physical experiment. These devices must be remotely operated through the network, and this may call for a variety of different technological solutions depending on the type of the equipment and the real physical experiment involved. Each laboratory setup -and often each associated learning scenario- may require a different type of human operation and control, raising considerable challenges particularly when performed remotely. Furthermore, from a didactical perspective, substantial effort is still needed for assessing the effectiveness of these learning modalities compared to traditional means of "hands-on" (on-site) laboratory training.

In this paper, we focus on the development of a virtual and remote laboratory platform for training in the operation and programming of complex mechatronic devices, such as robot manipulators. Studying the development of such virtual and remote laboratory modules in an educational programme, as an enhancement to theoretical course lectures, still constitutes a real challenge for the engineering education community; no standard solutions exist, and this includes both technological (remote operation and control) and educational (didactical/ pedagogical evaluation) perspectives. Indeed, there are very few attempts reported in the literature aiming to develop virtual and remote (web-based) laboratory systems in the robotics education field. One of these is described in [7], presenting a platform that includes, among other virtual (simulated) experiments, the control of a simple 2-dof robotic arm. This was based on a java applet performing kinematic simulation of the robot arm motion (with 2D only graphical animation). Simple motion commands can be issued at the joint trajectory level and can be used to convey basic principles of robot motion characteristics. The system illustrates basic web-based virtual laboratory concepts, but only in simulation (i.e. with no remote real robot in the loop). On the contrary, [8] presents a Java-based interface providing the functionality both to simulate and teleoperate a robot manipulator. This system can be thus used to practice movement commands of a simulated and/or remote robot manipulator, and can supposedly convey in a more efficient way the same basic concepts of robot motion control.

In comparison to these research efforts, we aim at developing a virtual robotic laboratory platform that will enable students to remotely practice real robot-manipulator programming tasks by making the most of the functionalities and programming modalities provided by the real robotic system. In other words, we want to offer students the possibility to learn how to program a real robot, without having one at proximity, in such a way that closely resembles the real robot programming operations and procedures. In this way, we can really refer to the platform as providing “distance training”, instead of a simple “familiarization” with robot motion principles. This is more clearly explained in the following section highlighting the learning objectives of the system. Section 3 presents the network and control architecture of our virtual robotic laboratory platform, together with all the technological design and development features. In section 4, we briefly describe the pilot experiment conducted to assess the performance of the system from an educational point of view. Initial results are discussed, followed by some concluding remarks and future work directions.

2 Virtual Laboratory Training Objectives

The general goal of our work is the development of a virtual laboratory platform to enable student training in robot manipulation and control technologies from any remote location via Internet. Robot manipulator arms and related mechatronic devices are not always readily available for experimentation by students in their training program. Access to such equipment for education and practical training purposes is often either limited by very specific time restrictions, or even not provided at all. Moreover, cost of such equipment makes it prohibitive for many academic institutes to obtain, and related laboratory training courses are completely missing from many educational curricula. Therefore, the benefits from providing a means for any-time/any-place (virtual and/or remote) experimentation in a “lab facilities sharing” context, are evident both from a socio-economic point of view, as well as directly related to the completeness and quality of practical training possibilities offered to students.

Existing virtual or remote laboratory systems, as already discussed above, are very few and provide some limited functionality in the sense of: (i) simulating and animating (in 2D or 3D) the motion of simple robot arms, (ii) practicing movement commands, which are usually issued either as desired end-effector’s position in -xy coordinates, or even directly as desired angles in the robot’s joint-space, and eventually (iii) submitting these commands for execution by a remotely located real robot. Such functionality can indeed demonstrate and teach students the basic principles of robot manipulation and control. However, programming a real robot arm to perform a specific manipulation task (e.g. a pick-and-place task in an assembly sequence) is usually somehow more complicated than that. The human operator should often resort in programming the task directly using the robot’s own programming language (usually some script-like interpreter language, such as VAL, V+ etc.); usually, however, an on-line robot programming scheme is employed, for instance using the robot’s Teach Pendant tool, in order to teach (record) the intermediate configurations that will constitute the complete robot motion sequence.

Taking into account these considerations, we directed our work towards the development of a virtual robot laboratory platform that will train students on how to program a robot manipulator arm, using the functionality and programming modalities provided by the real robotic system. The platform presented in this work incorporates a robot’s Teach Pendant emulator, as well as a virtual 3D robot animation panel integrated in the graphical user interface. The system enables students to create, edit and execute robot programs (i.e. complete motion sequences, such as a pick-and-place task), in exactly the same way as they would if they were using the real-robot's pendant tool. The programs can be viewed and edited by the student in simulation (with 2D and 3D animation features), and can also be sent for execution by a real robot manipulator, located in the premises of our robotics and automation laboratory. Other real robot programming modalities, even direct text editing and remote execution of program code in the robot's own programming language, could also be implemented, and are considered for integration in the near future.

Thus, the key issue to be emphasised is the support of real robot programming modalities within a virtual/remote laboratory platform, with the main objective being to provide students with realistic practical training on how to actually create and issue a complete robot manipulation program in a real-world task scenario. In this context of deploying a virtual/remote laboratory platform for robotics, our research is currently focusing on the following two main issues:

(a) From a *technological point of view*, we focus on the adaptation of concepts and technologies developed in the field of telerobotics, and on exploring their implementation in such remote laboratory settings. Robot teleoperation technologies have been constantly advancing and evolving for more than two decades now [9, 10]. Initial teleoperation systems were deployed in dangerous and hostile environments (e.g. in the nuclear industry for the telemanipulation of radioactive material). With the advent of communication and networking technologies, as well as with the development of new human-machine interactive simulation media (such as virtual reality systems [11]), research in the field of telerobotics has shown considerable progress, with new concepts proposed and demonstrated with success, such as "predictive displays" [12], "shared-autonomy" teleoperation control [13], or the "hidden-robot" concept [14].

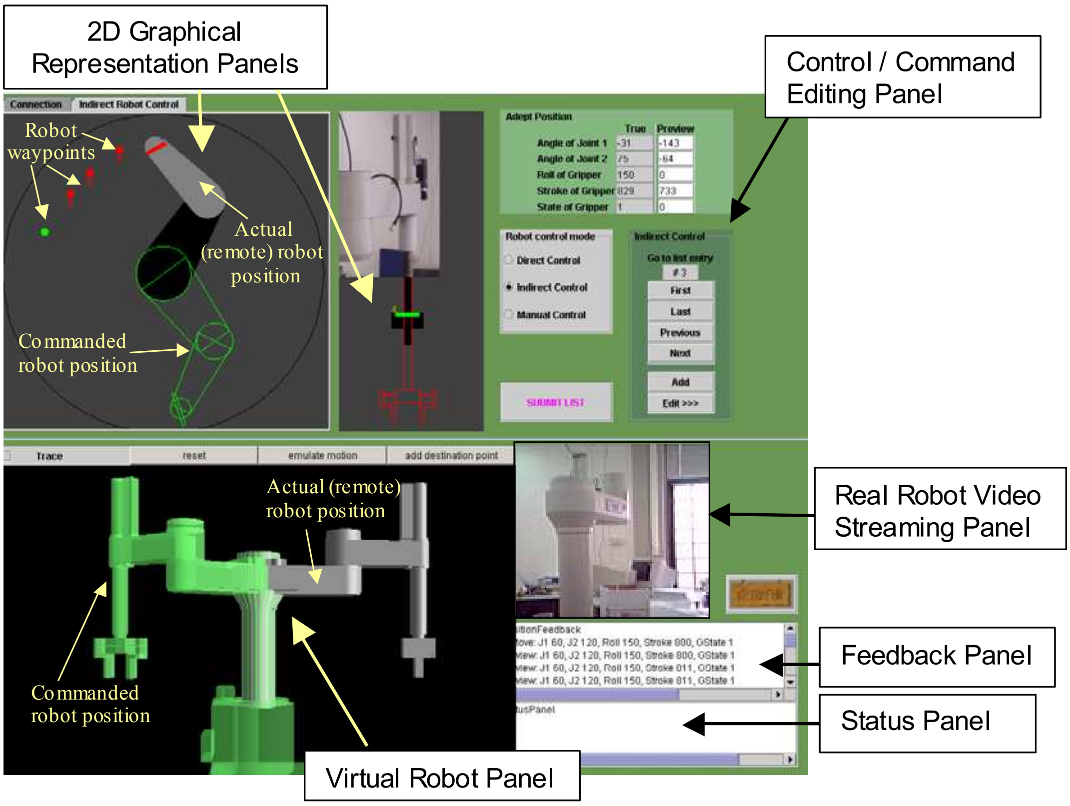
(b) From an *educational point of view*, teaching robot manipulation principles involves the familiarization with a variety of mechanical and control engineering concepts and skills. We aim to evaluate, principally from a pedagogical perspective, to which extent virtual laboratory scenarios can be effectively implemented in practice and used by students to obtain practical training as a supplement to theoretical courses. A literature review shows that the majority of the research results in this direction are restricted either in a qualitative type evaluation or in a "usability-oriented„ approach. On the contrary, we prefer to give emphasis on the didactical perspective in our evaluation approach, based on specific experimental protocols, combining qualitative and quantitative metrics.

3 Virtual Robotic Laboratory Platform

3.1 Web-based User Interface integrating Robot Teleoperation Concepts

The virtual robotic laboratory platform is developed based on Java technologies. The graphical user interface (GUI) integrates the following panels (see Fig. 1):

(1) *2D graphical representation* (top-view and side view) panels, visualizing both actual and commanded robot configurations,
(2) a *real-time video streaming* panel, which is based on RTP and implemented using JMF, showing (when on-line) the real remote manipulator in motion,
(3) a *control/command editing* panel,
(4) an interactive panel providing an exact emulation of the robot's Teach Pendant, called *Virtual Pendant*,
(5) *status and feedback* panels providing real-time textual information on current robot state, and
(6) a *virtual robot* panel, implemented using Java3D API, providing 3D visualization of both the commanded (preview animation) and the current robot configuration.



Adept Position

	True	Preview
Angle of Joint 1	-31	-143
Angle of Joint 2	75	-64
Roll of Gripper	150	0
Stroke of Gripper	829	733
State of Gripper	1	0

Robot control mode

○ Direct Control
● Indirect Control
○ Manual Control

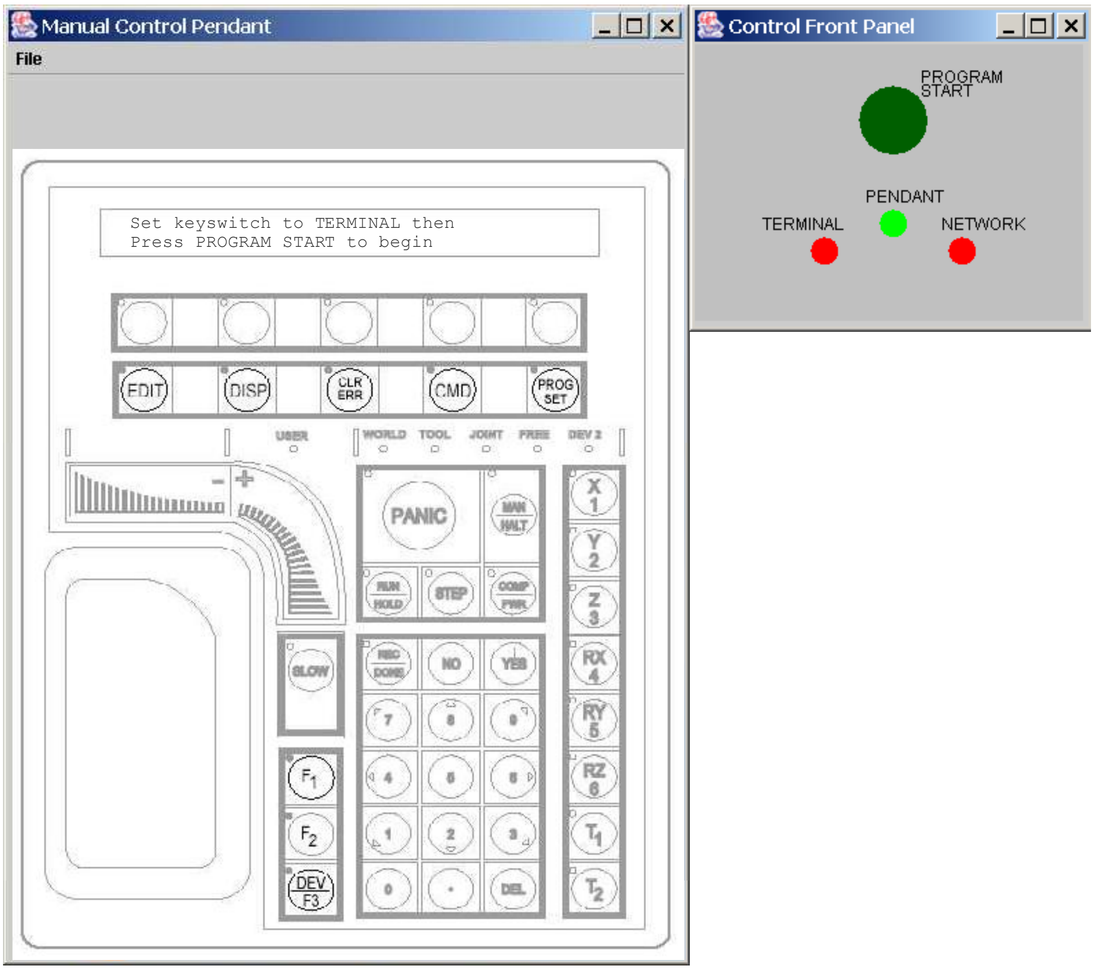
```
itionFeedback
ove: J1 60, J2 120, Roll 150, Stroke 800, GState 1
view: J1 60, J2 120, Roll 150, Stroke 800, GState 1
view: J1 60, J2 120, Roll 150, Stroke 011, GState 1
view: J1 60, J2 120, Roll 150, Stroke 011, GState 1
```

```
tusPanel
```

Fig. 1. The graphical user interface of the virtual robotic laboratory platform

The remote laboratory platform is based on a *client-server architecture*, providing the human operator with support of the following three robot control modes: (i) *direct teleoperation* control, (ii) *indirect control*, for robot *teleprogramming* via the command/editing panel, and (iii) *manual control*, that is, robot manipulator programming using the *Virtual Pendant* functionalities. These control modes are inspired from the telerobotics field, and particularly from work proposing various "shared-autonomy„ and "supervisory„ remote control modalities. In direct teleoperation, every command issued by the user (human operator) locally, i.e. within the GUI (master control site), is immediately transferred for execution to the remote (slave) robot. At the same time two types of feedback displays are active: (a) a *predictive display* (both in the 2D and 3D graphical panel) immediately visualising the commanded robot motion according to the human operator issued commands, and (b) a *real robot feedback display* (also both in 2D and 3D animation), showing where the robot actually is (that is, visualising current remote robot configuration, information provided in real-time through continuous feedback from the remote site).

As opposed to direct teleoperation, in the indirect "teleprogramming„ control mode the commands are generated off-line, queued in a list and submitted to the robot in a subsequent time frame, when the human operator decides to do so. The idea is to be able to create a complete robot program off-line, test its validity and optimality, before actually sending the command sequences for execution by the real robot. Based on the functionality (robot command editing routines, waypoints list creation etc.) of this indirect teleprogramming mode, we have developed a third "manual-control„ mode, which implements exactly the *Virtual Pendant* robot-programming scheme. According to our distance training objectives outlined in the previous section, the *Virtual Pendant* panel supports all the main functions of the real robot's pendant tool, and enables the student to learn and practice robot-programming routines locally. The user can create a robot program, add, edit or delete intermediate robot positions, as happens with the real robot's programming interface. He can then either



```
Set keyswitch to TERMINAL then
Press PROGRAM START to begin
```

Fig.2. An instance of the Virtual Pendant panel, providing an exact emulation of the robot's Teach Pendant functionality

"preview" the robot program visually on the 2D graphical representation panels of the interface, where an animation of the predicted robot motion is displayed, or "send" the program for remote execution on the real robot, and see the results of the actual manipulator motion on the video streaming panel (as well as on the 2D graphical panels that provide continuous position feedback to the user). An instance of the *Virtual Pendant* panel is shown in Fig. 2, where also a graphical representation of the robot's controller *Front Panel* is shown, containing buttons that the user must learn to manipulate in the correct order to initiate robot manipulation programming using the pendant.

3.2 System Architecture and Hardware Description

The graphical user interface described in the previous paragraph can run as an applet in any standard web browser, enabling users to connect via Internet (or LAN). Fig. 3 shows the overall client-server architecture of the virtual robotic laboratory platform. The system supports multiple connected users (terminal TE-1 to TE-n), through the implementation of a specific protocol using TCP/IP sockets for communication and real-time data exchange with the "*robot server*". Each client (student) can connect to the robot server either as an "observer", or as an "administrator", in which case (after entering the correct password) actual control of the real robot is obtained. Robot "observers" have access (through continuous data-feedback) to the current status and motion of the remote robot, while local (simulated) robot programming can also be performed. The robot administrator (only one logged-on at a time) has additional rights to send motion commands or complete motion sequences (robot

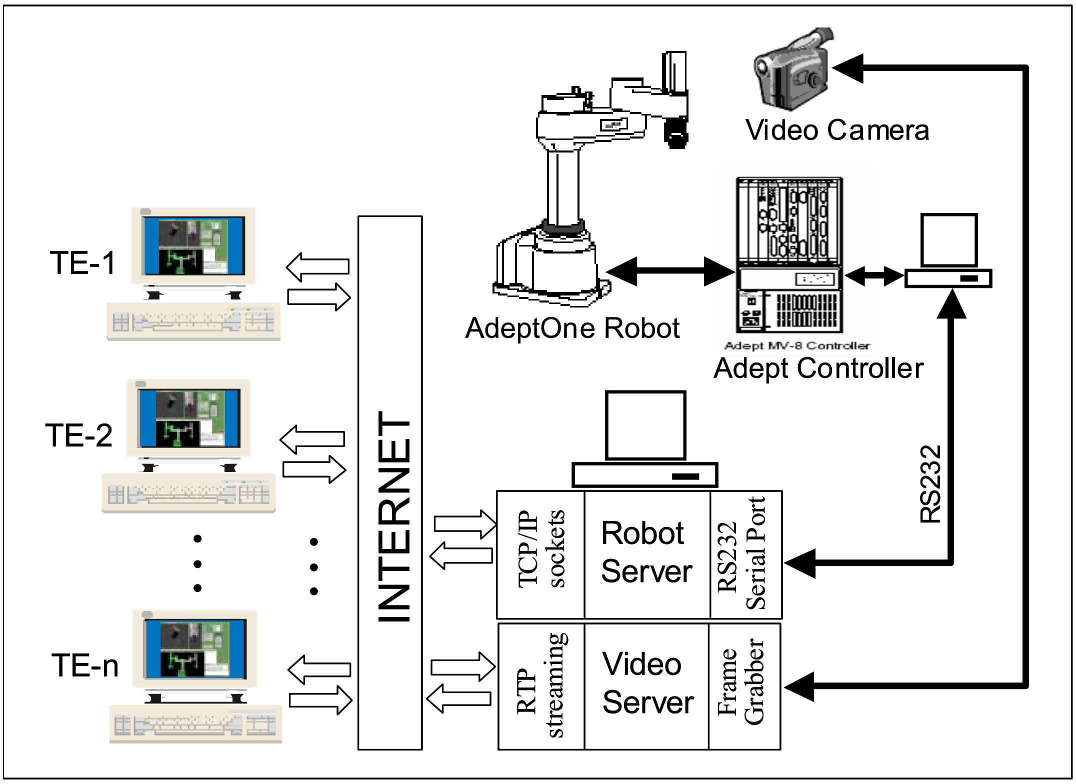


Fig. 3. Overall Architecture of the Virtual Robotic Laboratory platform.

program) to the remote robot manipulator for real execution. The robot server communicates with the Adept robot controller via an RS232 serial link, using an application specific protocol for real-time data exchange. In addition to the above, a separate “video server„ accepts calls from any remote location, establishing a direct video link that is based on RTP for real-time video streaming. In the current system configuration only one user can obtain real time video from the remote robot site. Multicasting has also been tested, but its potential application is limited, since it is usually not supported by any remote switching network.

The robot used in the experiments is a SCARA-type AdeptOne-MV manipulator, which has 4 degrees of freedom (3 rotational and 1 prismatic joint), as shown in Figure 4, and is also equipped with a pneumatic parallel-jaw gripper. From its kinematic structure, this robot manipulator is designed to perform planar motion profiles, and is therefore particularly suitable for assembly (e.g. pick-and-place) operations. The AdeptOne robot is programmed using the V+ robot programming language, which provides fast and real-time response, as well as multitasking capabilities.

4 Experimental Evaluation

We have conducted a pilot study on an initial version of the system (not incorporating yet the 3D “virtual robot” panel) to validate the usability of the system and in particular to assess its performance in terms of providing adequate “distance training” (regarding robot programming skills) to the students. In accordance with our objectives described in section 2, the key issue on which our research focused was the evaluation (principally from a pedagogical point of view) of the efficacy of the proposed virtual and remote robotic laboratory scenarios (in this case, programming of robot manipulation tasks). Our goal was to explore in which extent such distance training modalities can be efficiently implemented in

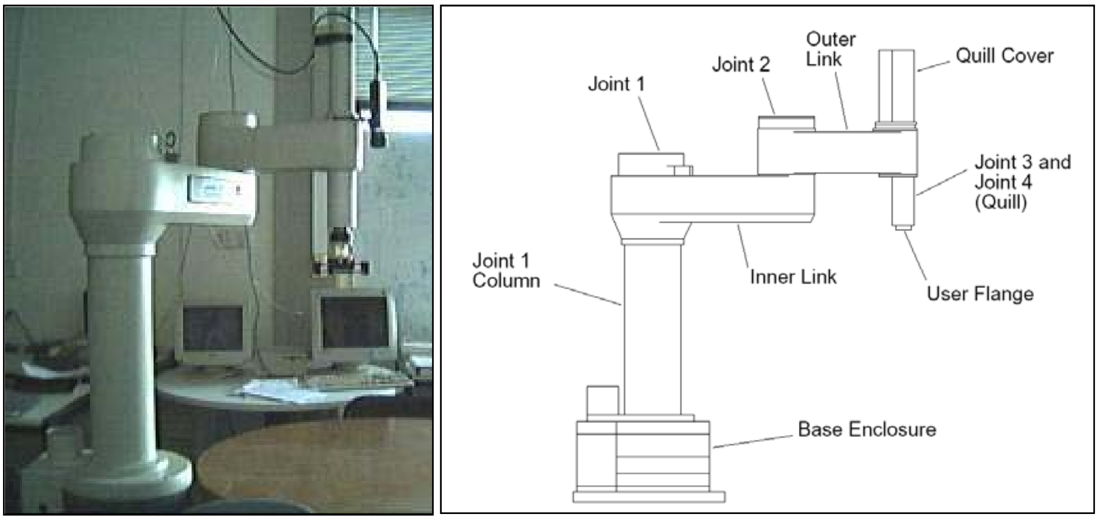


Fig.4. The AdeptOne robot manipulator used in the experiment and its kinematic structure.

practice, and used by students to obtain practical training as a supplement to a theoretic course track (in our case, an introductory course on robot manipulation).

We have designed a special *experimental evaluation protocol*, which was used consistently throughout the experiments. According to this protocol, the students participating in a laboratory training course (that complements a theoretical introductory course on robot kinematics, path-planning and control) were divided in two main groups: *group-I* trained the "classical way" on the real robot, while *group-II* was trained on the initial version of the user interface, using the remote laboratory platform as described briefly above. Both groups of students had undergone the same training phases and were exposed to exactly the same educational material by the trainer during each experimental session, with the only difference between the two groups being the direct contact (physical presence), or lack of it, with the real robot on-site. Both student groups completed their training session by conducting a specific experimental evaluation test on the real robot, where a robot programming task was assigned to them (namely, programming a pick-and-place operation using the real robot teaching pendant).

The test was sub-divided into distinct time phases, to facilitate tracking the performance of the students and identifying errors committed and/or difficulties encountered. Intermediate tests were also conducted (on the real robot or remotely using the telerobotic interface and the virtual pendant), in order to track differences in the learning curve between the two groups. A scoring chart was used by the trainer (tutor) during the experiment, and the errors were classified according to three main categories, namely: low-level technical skills, mid-level skills, and higher-level understanding, with different weights assigned to them. Teamwork between students (performing the experimental session in groups of 3-5 individuals) was also qualitatively monitored, while total time needed to complete each phase of the assessment test was also recorded by the tutor.

Based on the scoring chart and the associated penalty grades, a t-test of independent groups was followed in order to find out whether there exists statistically significant difference between the Means of the various test scores (low, mid, high, time and total) for the two groups (group-I: local and group-II: remote); group was the independent variable and score values were the dependent. Initial results show that there exist some apparent differences between the two groups for the three different score categories. Indeed, in the "low" category (representing errors committed related to low-level technical skills) group I (local) students made fewer mistakes compared to students of group II (remote). This could be explained by the fact that students forming the “local” group were trained the traditional way on-site, in physical contact with the real robot manipulator system, as opposed to group-II students who were trained remotely using the graphical user interface. Therefore, as it could be expected, group-I students exhibit a better “visual memorisation” of low-level technical dexterities, and thus better performance in the manipulation of the robot’s teach pendant. This is not the case for the mid- and high-level category skills, where the local group (group I) exhibited higher scores compared with the remote group (group II) (though differences proved to be smaller). This could be partially explained by the fact that students trained on a virtual environment appeared to have a better concentration and motivation level (as compared with students of the “local” group), which apparently aided them to assimilate higher-level concepts to a better extent.

However, it must be noted here that for all these quantitative results obtained during this first pilot study –both for the low and mid/high categories, as well as also for the total time and average score values– statistical analysis (t-test) reveals that all aforementioned differences in students’ performance are *non statistically significant*. Therefore, one can conclude that the remote laboratory platform with its graphical user interface described in this paper, created indeed a virtual training environment, which on its whole (integrating the various interactive control and visualisation panels) provided adequate learning elements, as related to mid and high level skills, compensating for the lack of direct physical presence on the real robot site.

5 Conclusion – Future Work

In this paper, we have described the development and experimental evaluation of a virtual laboratory platform aiming at the distance training of students in robot manipulator programming. Our research efforts focus on the adaptation of concepts and technologies developed in the field of telerobotics and virtual reality, and on exploring their implementation in such remote laboratory settings. The graphical user interface of the platform was developed based on Java technologies, and incorporates (among other control panels) a “Virtual Pendant” panel providing an exact emulation of the robot’s Teach Pendant functionality. The user interface also incorporates a “virtual robot panel” providing 3D visualization of both the commanded (in a preview robot animation mode) and the current robot configuration, supporting both direct teleoperation and indirect teleprogramming as remote robot control modes. A pilot experimental study with an initial version of the platform was conducted to assess the performance of the system in terms of remotely training students to program robot manipulation tasks with the Teach Pendant. The robot used in the experiments was a SCARA-type AdeptOne manipulator with 4 degrees of freedom. The experiments were conducted applying consistently a special evaluation protocol. Analysis of the initial results obtained from this first pilot study is encouraging, showing that this virtual laboratory concept can indeed be applied quite efficiently for training students remotely.

In the near future we are are planning to conduct a more thorough experimental evaluation study regarding: (a) the usability of the graphical user interface, in order to improve its design from an ergonomic point of view, and (b) the educational impact of the remote laboratory system, including also teaching robot programming involving the V+ programming language. Furthermore, we plan to explore how the integration of the virtual (3D graphics) robot panel can ameliorate the realism of the simulation in an off-line self-education mode, and increase both its user-friendliness and educational impact upon students.

Acknowledgement

This work was partially supported by the Greek General Secretariat for Research and Technology, and the European Commission in an international Greek-German cooperation framework.

References:

[1] B.A. Foss, K.E. Malvig, and T.I. Eikaas, “Remote Experimentation - New Content in Distance Learning“, In Proc: *International Conference in Engineering Education* (ICEE’2001), Session 8D1, Oslo, Norway, August 6-10, 2001.

[2] T.I. Eikaas, B. Foss, “CYBERLAB - A Global Remote Laboratory Experimentation Network“, *AIChe 2003 Annual Meeting*, Session 501: New Technologies For Experimentation Over The Internet, November 17, 2003, San Francisco, CA.

[3] “Emersion – a New Learning Technologies Project. Hands-on resources for flexible learning in engineering education”: http://elearning.epfl.ch/emersion.html

[4] T.A. Fjeldly, J.O. Strandman, and R. Berntzen, “LAB-on-WEB: A Comprehensive Electronic Device Laboratory on a Chip Accessible via Internet”, in: Proc. *International Conference on Engineering Education* (ICEE 2002), Manchester, UK, August (2002)

[5] Dabney, J.B., Ghorbel, F.H., and McCune, J., "Web-based control of the Rice SPENDULAP," *International Journal of Engineering Education*, Special Issue on: ‘Distance Controlled Laboratories and Learning Systems’, Vol. 19, No. 3, pp. 478 - 486, 2003.

[6] J.Henry and C.Knight, “Modern Engineering Laboratories at a Distance,” *International Journal of Engineering Education*, Special Issue on: ‘Distance Controlled Laboratories and Learning Systems’, vol. 19, no. 3, pp. 403-408, 2003.

[7] M. Karweit, “A Virtual Engineering/Science Laboratory Course,” Johns Hopkins Univ., Dept. Chemical Eng. VIRTLAB project: http://www.jhu.edu/%7evirtlab/virtlab.html

[8] F. A. Candelas, S. T. Puente, F. Torres, F. G. Ortiz, P. Gil, J. Pomares, "A Virtual Laboratory for Teaching Robotics," *International Journal of Engineering Education*, Special Issue on: Distance Controlled Laboratories and Learning Systems, vol. 19, no. 3, pp. 363-370, 2003.

[9] J. Vertut and P. Coiffet. *Les Robots: Téléopération. Tome 3A: Evolution des technologies. Tome 3B: Téléopération assistée par ordinateur.* Edition Hermes, Paris, 1984

[10] T. B. Sheridan. *Telerobotics, Automation and Human Supervisory Control*. The MIT Press, Cambridge, USA, 1992

[11] G. Burdea and P. Coiffet. *Virtual Reality Technology*. John Wiley, 1994.

[12] A. K. Bejczy, W. S. Kim, S. Venema, “The Phantom Robot: Predictive Displays for Teleoperation with Time Delay”, *1990 IEEE Int. Conf. on Robotics and Automation* (ICRA’90).

[13] B. Brunner, G. Hirzinger, K. Landzettel, J. Heindl, “ Multisensory shared autonomy and tele-sensor-programming - key issues in the space robot technology experiment ROTEX“, In Proc: *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* (IROS’93), Yokohama, July 26-30, 1993.

[14] A. Kheddar, C. Tzafestas, P. Coiffet, T. Kotoku, K. Tanie, “Multi-Robot Teleoperation Using Direct Human Hand Actions”, *International Journal of Advanced Robotics*, Vol. 11, No. 8, pp. 799-825, 1997.

Author(s):

Contact Author:
Dr. Costas S. Tzafestas, Lecturer,
National Technical University of Athens (N.T.U.A.),
School of Electrical and Computer Engineering,
Division of Signals, Control and Robotics,
Zografou, Athens 15773, Greece.
Tel (office): +30-210-7723687
Fax: +30-210-7722490.
Email: ktzaf@softlab.ntua.gr

Mr. Manthos Alifragis, Ph.D. Student,
National Technical University of Athens,
School of Electrical and Computer Engineering,
Division of Signals, Control and Robotics

Dr. Nektaria Palaiologou,
University of Piraeus,
Department of Technology Education and Digital Systems,
18534 Piraeus, Greece.
E-mail: npal@unipi.gr

Dr. Stelios C.A. Thomopoulos,
Institute of Informatics and Telecommunications,
National Center for Scientific Research "Demokritos",
15310 Athens, Greece.
E-mail: scat@iit.demokritos.gr

Mrs. Athanassia-Elena Exarchou,
IS Integrated Product Information (K-DOE-5), VW Group, Wolfsburg, Germany.
E-mail: athanasia-elena.exarchou@volkswagen.de

Mr. Alexander Kroys and Mr. Rüdiger Kunicke, Graduate Students,
Otto-von-Guericke University Magdeburg,
Faculty of Computer Science, 39106 Magdeburg, Germany.