

Using Robot Programming to Foster Learning Capability

Aurilla Aurelie Bechina

Buskerud College University
Pb. 235, 3603 Kongsberg, Norway
Aurillaa@hihm.no

Dag Samuelson

Buskerud College University
Pb. 235, 3603 Kongsberg, Norway
Dag.Samuelson@hibu.no

Tord Fauskanger

Respons data, Kongsberg, Norway
tord.fauskanger@gmail.com

Abstract- There is growing need in programming capability. However, recent studies in Norway highlights that there is a rather a lack of available competences in programming skill in the job market. One reason is due to the lack of enthusiasms of students to undertake studies in computer programming. One way to overcome this issue is to make this kind of education more appealing to high school student by developing an online remote educational system. The aim of our project is to provide a motivating environment to learn programming by moving a small robot placed in a room at the University lab. Students will learn to program by testing in real time the movement of the robots. The paper discusses how a remote programming lab can foster learning capability by providing an entertaining environment. After introducing some concepts of constructivist learning, the paper presents a prototype of the learning solution. Based on an assessment of the usage of this remote programming solution, the paper discusses the issues and concepts of such learning system.

Keywords- Mobile Robot, remote laboratory, programming platform

I. INTRODUCTION

The numbers of students attracted to computer science engineering is sharply decreasing. For example, a 2003 Harris poll conducted in the U.S. shows that only 2 percent of first-year university students (and less than 1 percent of young women) want to major in computer science [1]. This alarming trend is as well seen in Norway. The last couple of years, the Norwegian IT employment marketplaces are struggling to find qualified software engineers. Even, despite the world financial crisis, Norwegian companies kept hiring some students that did not even yet complete their Master degrees in software engineering. The computer Science student's enrolment in higher Education has been quite inconsistent and quite low compared to need of employment sectors. There are many reasons for the lack of interest in pursuing education within the engineering field [2]. Several research studies highlight that many students have negative or positive stereotypes about engineers (e.g. engineers are nerds or have to be genius), or have no clear pictures of what engineers tasks are [3].

One another recurrent reason that is often invoked in the lack of interest to undertake software engineering education is due mainly to the perceived difficulties of this type of

study [3]. In addition, many students have issues connecting the theory learnt in class with the practical use of it, thus creating a dislike in programming.

Therefore, there is a stringent need to promote the software education and to motivate the students to take programming classes. Practical exercises and case studies are recognized to be a good approaches foster learning capability. For example, lab experiments have played an important role in the curriculum for engineering students.

It is well recognized that Lab experiment contribute to facilitate the learning and this even more valid for programming skill. Our project aims at investigating how learning to program a mobile robot will enhance the learning of programming skill. The entertainment part keep the students focusing on programming for example the move of the robots.

Traditional labs exercises are done in a laboratory within the premises of the college or university and will often require a lot of preparation from the teacher. In addition, the cost of the equipment and the set up can be quite high. Therefore, in our study we intend to use the concept of remote laboratories allowing remote execution of Lab experiment.

This paper discusses how college and university could introduce a certain level of innovative approach in the way teaching and learning processes are performed. We intend to investigate how programming remotely a mobile robot can foster the learning capability in programming of students.

A prototype named **REPOR (Remote Educational Programming Of Robots)** has been developed. An assessment of the usage of the remote programming system amongst students has been performed. The data collection is based on both qualitative and quantitative approaches.

The next section, presents the concept of remote lab, a brief state of Art is reviewed. The section three presents the prototype that

has been developed. The section four outlines the results of the conducted assessment. Finally, section five provides some concluding remarks.

II. REMOTE LAB

Traditional labs require traditional lab equipment that can either be expensive or quickly outdated. Due to budget constraints, Higher Education has limited amount of equipment and therefore students might have to share the experiments or wait their turn. Access to this lab is as well quite limited and requires the presence of a teacher. Therefore, the number of experiments in educational students is quite often decreasing, thus creating a less interest from students in the engineering field. In our research study, we intend to use a remote Lab as a means to overcome the issues stated above. A remote lab is a laboratory that can be accessed and managed from a remote location [4]. The lab includes physical components that are considered as resources. The monitoring of the experiment of the Lab is done through a Web-Camera. Thus, the students can visualize in real-time the status of the experiment. If the experiment is batched type, video stream is recorded and can be viewed when the experiment is finished. The main interest of Remote lab relies on the fact that it can be shared amongst several users, provided that the management software can handle the scheduling. This allows an extremely important feature for the overall systems as it can be considered as a Learning Management System (LMS) supporting access anytime, anywhere to the students with the right access [5].

In our project, we have decided to use robots as it has been recognized by the educational body that they can play a strong role in the fostering learning capability. Usage of Robots and robotics involves several technical fields ranging from mathematics, Physics, electronic to software engineering. Robots and robotics have been used in engineering education as a motivating factor to teach students various

subjects [6]. Programming skill is important in the robotics field; therefore students are required to learn if they want to play with robots. Han and Gao stated in their study that 64 % of the students reported that they learnt to program by performing robotics exercises [7]. This is a good indicator that robotics can serve as a good medium for learning programming.

Robots provide combined features of learning while playing. Game based Learning has proved to be a useful tool to educate students [8-10].

The last decade, increasing number of solutions have emerged such as CYBERLAB [11], Virtual Laboratory in Mechatronics: Access to Remote and Virtual eLearning (MARVEL) [12], LABNET [13], NetLab [14], iLab [15] and so forth. The review of these solutions indicates that most of them were intended for one specific scenario: meaning without the possibility to reuse some of the developed components. In order to develop our prototype REPOR, we decide to choose an open solution iLAB from MIT. The iLab architecture consists of multiple services that are interconnected by a service broker, the iLab service broker.

The user-side scheduling service (USS) handles the users side of the scheduling. The lab-side scheduling service (LSS) handles the lab aspects of the scheduling. The experiment storage service (ESS) handles the storage of experiment data. And last the experiment service handles the end point for access to experiments

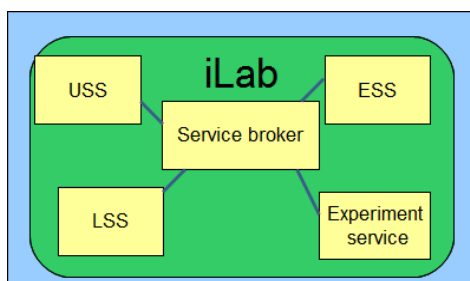


Fig. 1 Overview of iLab

III. REPOR Architecture

The project is unfolded in two subprojects. One part concerns the realization of the construction of the robot itself. Adequate physical components needed to be selected and assembled. For the purpose of the simplicity of the test, we have used the Lego NXT. The second Bugster is still under construction by the students. The NXT robot can be modified to run Java and is easily adapted to the system through use of the template system.

leJOS is a replacement firmware for the NXT Lego mindstorms robots. The robots are based on a Java stack, with extensive robotics API. The robotics API contains navigation, servo control, simple artificial intelligence and a lot more

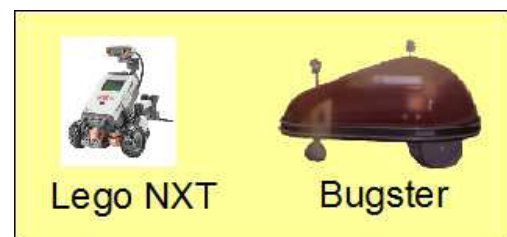


Fig. 2 Selected Robots

The second part of the project concerns the software application that will manage the access and the control of the robots. The robot use is meant to be shared by several students based in different location. Therefore, remote access functions are important requirements. Hence, students have the possibility to schedule a time slot for robot use. Another requirement for the overall system is that it should be easy and provide good interoperability amongst various platforms. Based on technical and users requirements, a general architecture pictured in figure 3 has been delineated.

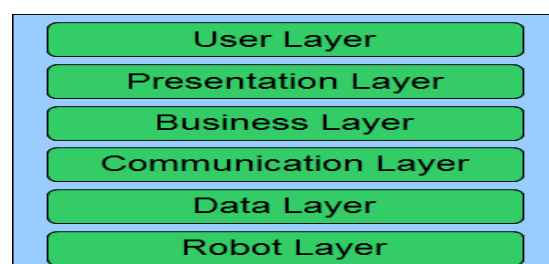


Fig. 3 Different layers of the system

The architecture encompasses 6 layer models. At the top is the user Layer containing different user's roles such students, teachers and administrator.

The presentations layer contains all the visual components which the users can interact with. For reusing purpose and avoiding re-programming everything from the scratch, we used the web interface of iLab and the lab client interface.

The business layer holds most of the business logic in the system such as management of the system, work stored in database and so forth.

The communications layer defines all of the different communication protocols that the system will use. The web interface uses HTTPS to securely communicate with the user. We used the communication provided by iLab such SOAP messages.

The data layer encompasses data dependent components needing access to a database to store persistent data.

Finally, the robot layer concerns the building of the user software. The robot layer contains user software, robot templates, monitoring component and so forth.

The robot environment is pictured in figure 4

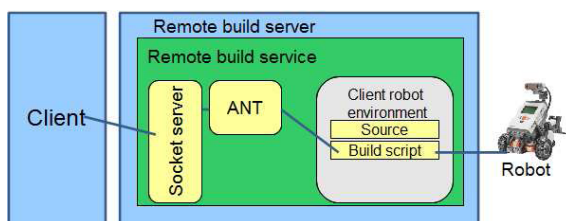


Fig. 4

The various platform used in this project are based on Microsoft. The prototype is up and running and has been tested. The components have been individually tested. The overall system went as well thought a carefully testing phase.

IV. REPORT EVALUATION

After the test of the prototype, the next step was the evaluation of the use of the remote lab REPOR by students of Buskerud

College University. Although, it is still ongoing research work, it is important to conduct this assessment in order to either implement new functionalities or to improve the existing ones. For example, it is crucial that the User Interface is sufficiently intuitive and suitable for this kind of remote Lab. In order to investigate how the students were using the programming environment and to detect any issues, we adopted two approaches.

Qualitative methods' based on observation of students while using the system, followed by interviews. This approach proved to be useful to gather important information such as the facility to use the environment, the behavior of the student, the way he was programming the movement of the robot, and so forth.

Quantitative method based on the design of questionnaire that we submit to several students that have already tested the remote Lab system to learn how to program via the remote programming of a mobile robot. In this survey, we gathered information about the perception of the usability of the REPOR system in improving their programming skill. Questions were as well related to the interest of these remote programming verse traditional lectures in programming. It was important as well to evaluate the level of satisfaction of students in using the REPOR system. The questionnaire encompasses 10 close questions to 30 students. The students were selected from third and fifth year engineering students.

The analysis of the collected data indicates that most of the students did like the idea of using remote Lab for learning to program. They highlight that the fact to use the system while being outside the school was as well beneficial. The REPOR system encourages them to do extra effort to understand why they program did not work in some cases. The linkage between their program code and the moves of the robot help them to understand better some of the programming functions in Java. However, due to the limited functions of the current

prototype, there were some minor bugs that hamper the use test of the overall system by the students. Overall the students were quite positive with this experiment.

The assessment presents some limitations such as low number of students testing the systems and the need to further develop the REPOR system.

V. CONCLUSIONS

This paper has presented a prototype of a remote Lab. The educational environment is composed of mobile robots in a lab of the College University, Web camera, and software management accessible via internet.

REPOR is based on iLab that is suitable for reuse, and can be easily adapted to specific requirements and to support additional technologies. For example, we have added full support for interactive applets.

The students can program remotely the moves of the robots via a web Browser. Access to the systems is regulated. By using the user interface, the students can test their programming skill by seeing the direct implication of their program code in the move of the robots. This is still an ongoing research project and is under further development. The prototype has been evaluated and the first results are encouraging. Analysis of the collected data conclude that using a remote lab programming in order to learn to program is particularly welcomed by the students. As a motivational tool, the students put more efforts in learning the programming which is subject considered to be less appealing.

The positive attitudes of the students toward the overall environment and the results of students lead us to believe that providing an alternative approach to learn programming such as playing while learning could probably influence the students to choose more easily engineering field.

This prototype is still under development and therefore, based on the preliminary assessment, we aim at improving the user

interface and developing further technical functions. These features include for i.e multiple users, means of real-time communication or cooperation between the teacher and the students. Collaborative communication tools e.g. chat, video could be integrated in to the lab framework, and in the lab clients; thus, enhancing the learning capability by sharing insights and experiences.

The learning capability based on constructivism learning should be further investigated.

REFERENCES

- [1] IEEE. *Innovation in Danger: Not Enough Students Pursuing Engineering Careers*. 2003; Available from: http://www.ieee.org/portal/cms_docs_iportals/iportals/aboutus/news/2008/education_press_release.pdf.
- [2] BURTNER, J., *The Use of Discriminant Analysis to Investigate the Influence of Non-Cognitive Factors on Engineering School Persistence*. Journal of Engineering Education, 2005.
- [3] Gibbons, S.J., et al., *Middle school students' attitudes to and knowledge about engineering*, in *International Conference on Engineering Education*. 2004: Gainesville, Florida.
- [4] Šafaric, R., et al., *Control and robotics remote laboratory for engineering education*. International Journal of Online Engineering, 2005. 1(1).
- [5] Mlitwa Nhlanhla, B.W., *Technology for teaching and learning in higher education contexts: Activity theory and actor network theory analytical perspectives*. International Journal of Education and Development using Information and Communication Technology, 2007. 3(4): p. 54-70.
- [6] Garner, J.R., et al. *The Remote Exploration Program: a collaborative outreach approach to robotics education*. in *Robotics and Automation, 2004. Proceedings. '04. 2004 in International Conference on Robotics and Automation*. 2004.
- [7] Han, J. and R. Gao, *Motivation and Training Program on Robotics Education*, in *International Conference on Computational Intelligence and Software Engineering, CiSE IEEE*, Editor. 2009, IEEE. p. 1-4.
- [8] de Freitas, S. and M. Oliver, *How can exploratory learning with games and simulations within the curriculum be most effectively evaluated?* Computers & Education, 2006. 46(3): p. 249-264.

- [9] Hainey, T., et al., *Evaluation of a game to teach requirements collection and analysis in software engineering at tertiary education level*. Computers & Education. **In Press, Corrected Proof**.
- [10] Moreno-Ger, P., et al., *Educational game design for online education*. Computers in Human Behavior, 2008. **24**(6): p. 2530-2540.
- [11] ciberlag. 2009; Available from: http://www.cyberlab.org/wp/wordpress/?page_id=2.
- [12] Müller, D., et al., *Mixed reality learning spaces for collaborative experimentation: A challenge for engineering education and training*. International Journal of Online Engineering (IJOE), 2007. **3**(4).
- [13] Aliane, N., *LABNET: A Remote Control Engineering Laboratory*. International Journal of Online Engineering (iJOE). **3**(2).
- [14] Z. Nedic, J.M., and A. Nafalski,, *Remote Laboratories Versus Virtual and Real Laboratories*, in *33rd ASEE/IEEE Frontiers in Education Conference*. 2003: Bolder, USA.
- [15] iLab , M. *ILab*. Available from: <http://ilab.mit.edu/iLabServiceBroker/>.