Interactive Lab Experimentation and Simulation Tools for Remote Laboratories

Fariba Moghaddam[1[0000-0001-6800-7776]], Denis Gillet[2[0000-0002-2570-929X]]

Aldo Vaccari[1[0000-0002-9591-5283]], Christophe Salzmann[2[0000-0002-1572-252X]]

Yves Piguet[2[0000-0002-8725-6601]]

[1] University of Applied Sciences and Arts Western Switzerland, 1950 Sion, Switzerland
[2] École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland

Abstract. New extensions and features to the remote laboratories currently used for undergraduate courses in automatic control laboratory at the School of Engineering of University of Applied Sciences and Arts Western Switzerland are presented. This paper shows how lab experimentation and lab simulation can be linked in a complementary way. In fact, their adequate combination offers the students the opportunity to apply classroom theoretical knowledge while conducting a real experiment. Besides, the lab-based simulation tools can be used for experiment modeling and controller design. Simulation tools also facilitate analyzing and interpreting data and finally validating the results. We discuss different technical implementing trends as well as several pedagogical approaches in the way remote lab exercises are performed including real-time data acquisition, experimentation environment within user friendly interface and simulation tools.

Keywords: Remote Labs , Simulation Tools, Smart Lab Experiments, Real Time, Learning Analytics.

1 Introduction

In science and engineering education, it is imperative that students be able to put into practice what they have learned in the classroom and to make the transition from theory to practice with confidence and effectiveness. If hands-on laboratories have long played a central role in engineering education, financial and practical constraints can seriously limit access to these activities. The importance and flexibility of distance learning, whereby teaching is undertaken remotely on digital platforms, have been largely proven, especially during the Covid-19 pandemic. Remote labs provide access to scientific experiences through a digital learning platform where students can interact with the real experiments not locally available.

There are two different ways to conduct laboratories: experimentation and simulation. They both play an important role in engineering education. Students feedbacks show that they consider remote labs as being as operative as hands-on labs. Furthermore, they seem to think that simulations are less effective than remote labs because simulations don't feel as realistic. Indeed, the realistic nature of remote labs affords students the opportunity to more directly apply theories learned in the classroom to real-world, while simulations may lead students to overlook and even to miss the link between theory and application.

The goal of this work is to show how lab experimentation and lab simulation can be linked in a complementary way. In fact, their adequate combination offers the students the opportunity to apply classroom theoretical knowledge while conducting a real experiment. Besides, they can use the lab-based simulation tools for experiment modeling and controller design. Simulation tools also facilitate analyzing and interpreting data and finally validating the results.

This paper presents the remote experimentation environment together with interactive simulation tools used for automatic control laboratory sessions at the school of engineering of University of Applied Sciences and Arts Western Switzerland. We discuss different technical implementing trends as well as several pedagogical approaches in the way remote lab exercises are performed including real-time data acquisition, experimentation environment within user friendly interface and simulation tools.

2 Laboratory Experiments for Automatic Control

We suggest a classic control laboratory whose aim is to offer remote users the ability to design PID (proportional-integral-derivative) controllers and test them for example on a servomotor for speed and position control. A DC (direct-current) motor is paired with an incremental encoder to provide position and speed feedback.

The experiment remote control panel is a user-friendly interface which allows real-time data acquisition and visualization of the physical measurements such as speed, position, voltage on the remote servomotor, in open-loop and closed-loop configurations (Fig. 1). A camera offers continuous live video streaming of the experiment in addition to other remote measurements. It has been shown that the degree to which students derive realism from the remote experience is certainly a function of the user interface and visualizations used in the lab experiences.

Real-time Interaction over Internet should enable students to efficiently interact with physical distance equipment, as a tight interaction provides them with the best possible feedback to minimize drawbacks inherent to the distance, with sufficient information to reproduce the state of the distant equipment and its operational conditions.

Fig. 1. Remote control panel for a servomotor speed and position control

Besides, the simulation environment is integrated to the same remote control panel application where the students interact with the remote experimentation system in real time (see section 3. Interactive Tools).

The experiment remote control panel together with the simulation tools propose the following sequence of learning objectives:

1. Open-loop experimentation on the remote system to characterize its dynamic behavior, for example by performing a step response,
2. Data collection and data processing,
3. System model building through interactive curve fitting to identify the transfer function of the system,
4. Controller design based on the identified model and other given specifications,
5. Closed-loop tests to validate the calculated controller on the remote system for different set points,
6. Controller fine-tuning based on closed-loop simulation and experimentation tests,
7. Disturbance insertion on the real system and test the robustness of the controller,
8. Observation and interpretation of the results.

The proposed control panel is implemented in a MOOC platform and is actually fully dedicated to hands-on activities. From a pedagogical point of view, one deals with the extensions of e-Infrastructure for self-organized learning environments. Different learning scenarios and pedagogical contents split into theoretical, experimental and evaluation phases lead the student during the laboratory session. Based on pedagogical contents, students are able to get orientation on their own, as well as to determine and elaborate adequate learning methods for new competences and skills. The learning scenario of each session is split into short videos describing how to use the various tools (experiment description, theoretical development, control panels, …), a set of offline exercises and a set of numerical questions (quiz form) to validate user's finding and to evaluate user's understanding.

3 Interactive Tools

In addition to the control panel where students can set up the experiment conditions, supervise its proceeding and collect data, software modules provide computation and visualization support for the design of feedback controllers. Interactivity invites students to investigate different hypotheses, but without guiding them too tightly: students remain free to also try faulty designs and learn from the mismatch between their expectations and their observation of simulations and experiment results.

3.1 Purpose

The typical steps to obtain a suitable controller are as follows:

1. System modeling. Among the mathematical tools used in automatic control, transfer functions, which related the system output (measurement) to the system input (control signal) and assume linearity and time invariance, are an extremely powerful approach. They can be obtained from first principles, decomposing the system into subparts, each with their dynamic behavior described by differential or algebraic equations, with Laplace transform counterpart; or from the parameter identification of "black-box" generic models. Often both approaches are used together: a qualitative understanding of the system provides the model structure (model order, integrator or chain of integrators, delay) with unknown lumped parameters which are identified from input-output experiment data.

   Simple first- or second-order models, possibly with an integrator or a time delay, have a very simple step response students must get familiar with. In the system modeling tool, a sequence of system input and output samples acquired from the remote experiment are displayed. The student can select a range of samples where the effect of the steady-state gain and the time constant(s) are clearly visible, and then fit asymptotic lines whose position gives the model parameters. The resulting model response is displayed along with the samples to show the quality of the fit.

2. Controller design. Depending on the system, on the available measurements (system output) and actuators (system input), a controller structure must be chosen. The structure can take into account system knowledge which is not captured in the model, to enforce high or low gains in some frequency ranges and reject perturbation and measurement noise. In the case of a servomechanism, dry friction, a non-linear perturbation which can be approximated as a constant drag when the velocity is not reversed, can be rejected with an integral effect. Once the structure is decided, controller parameters must be adjusted to obtain

the desired closed-loop behavior in terms of stability, robustness and performance. To achieve a suitable compromise between contradictory goals, these criteria are evaluated simultaneously and displayed as simulations in the time domain and as frequency responses. With direct manipulation, students get a better understanding of how they are related, what can be obtained, and why they are useful.

3. Controller validation on the remote experiment. The controller which has been designed and analyzed with computed graphics can be applied to the real system. If there is a good match, students get confidence in their approach. Otherwise, the comparison triggers reflection to understand the causes of the discrepancy. Was the model suitable for its intended use? Does the real system have additional aspects which were neglected, for instance saturation of the input, or the effect of sampling or noise?

3.2 User experience design

The interactive tools were added to the existing user interface for our remote experiments (Fig. 2). This user interface is implemented as a web application to avoid any installation for the students. It shows the block diagram of the system, either in open loop without a controller or in closed loop with the controller. The diagram structure, the input shape and parameters, the controller structure and parameters, and the sensors to use can be selected easily with buttons, pop down menus and numeric input fields. All the settings remain visible to let the students know exactly what they are doing and to save their work with screenshots, by saving the web browser page, or printing to PDF or to a real printer. The display is especially important when two students collaborate, possibly remotely where they cannot see what their colleague does with the mouse or the keyboard, or when they request advises from an assistant. A video stream shows the experiment and provides an easy way to understand what happens. For the servomechanism, measurements above the noise level and in a suitable bandwidth are clearly visible.

Interactive tools are added below the area reserved to the remote experiment, in different tabs. Tools correspond to different steps, as described in the previous section. Each step uses data from the previous one: acquisition sampled data from open-loop experimentation to modeling, model parameters from modeling to controller design, and controller parameters from controller design to validation on the remote experiment. Data can be saved to the local file system, which serves three purposes: students can check their content and understand clearly what happens, without hidden steps obscured by the software; they can reload data in the same tool or for the next step, maybe coming back at another time after they have thought “offline” about their work; and they can import data into other applications, such as Matlab or a word processor, for further analysis or to write a report.

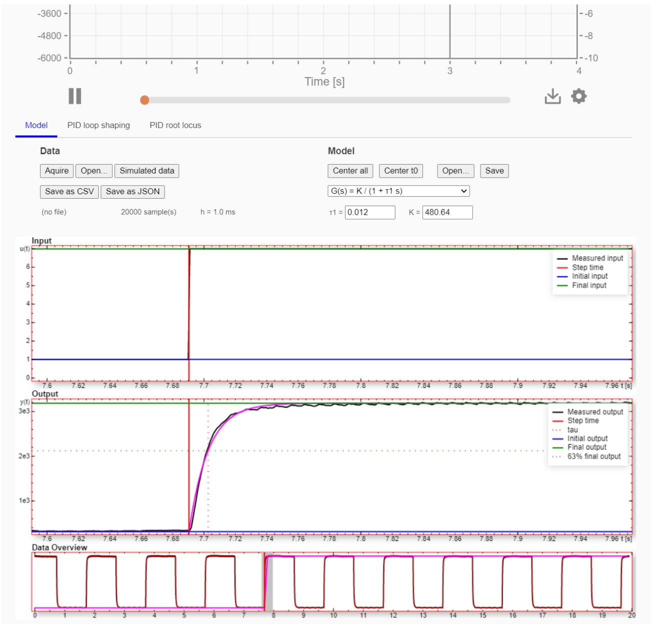

Fig. 2. Interactive tool for transfer function fit on experimental responses


```
{
    "K":480.63537749401445,
    "tau":0.012237174850223909,
    "integrator":false,
    "d":null
}
```


Fig. 3. Outcome of transfer function fit as a JSON file

The main exchange format is JSON, a text format easy to read and expressive. Fig. 3 shows the model parameters corresponding to the model fit of Fig. 2. Samples acquired from the experiment can also be saved to and loaded from CSV files which are more suitable to tabular data.

3.3 Technical aspects

The interactive tools have been developed in Sysquake, a scientific application for numerical computation and interactive visualization with a programming language mostly compatible with Matlab. Sysquake exists in two versions, native for optimal computation performance, and web-based to run locally in the browser without server support. The same interactive applications run likewise in both versions.

For the simulation tools, applications were compiled to JavaScript and integrated into the HTML page which contains the remote experiment control panel. HTML input elements, such as pop down menus and input fields, were added to permit direct parameter entry in addition to the direct manipulation of graphics, and import/export functionality was added to interface with the remote experiment on one side to fetch experimental data or set parameters, and with files on the other side to save or reload data and controller designs. Because of the time allocated to students for the remote lab experiments, this approach was preferred to a command-line interface which would be more open to different kinds of analyses and controller design methods, but would requires more learning efforts.

4 Typical Student Session

While students can freely switch between the remote experiment and the interactive tools, the expected typical path they would take could be as follows:

- Set the parameters in the remote experiment control panel to have a periodic square signal with an output significantly larger than the measurement noise, and a period significantly larger than the system time constant.
- Acquire samples in the identification tool.
- Select samples around a clean step response.
- Fit a model of adequate structure, typically a first-order model for the servomechanism controlled in speed (Fig. 2).
- Save the model to a JSON file, switch to the controller design tool, and load the model (Fig. 3).
- Design a controller with good robustness and performances. For the servomechanism controlled in speed, a PI controller (PID without derivative term) is expected to give good results. The phase margin can be observed in the Nyquist diagram, the step overshoot in the step response of the system output, and the control signal (system input) can be made smaller than the servomechanism input voltage admissible range (Fig. 4).

The controller parameters are applied to the remote experiment, which is switched to closed-loop mode. The setpoint signal can be changed to obtain a suitable response (Fig. 5).

5 Trends in Implementing Remote Labs

While remote experiments have been deployed for more than two decades, they are based on technologies which have evolved continuously [1]. Three aspects are worth mentioning: The seamless integration in learning platforms, which was described in previous section; the changing optimal compromise which results from processing power available on the server and the client and the communication bandwidth; and security aspects.

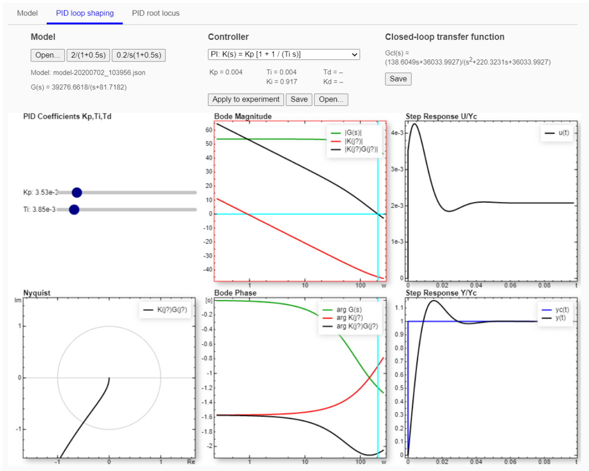

Fig. 4. Design of a PI controller with the interactive tool. The controller frequency response can be manipulated in the magnitude Bode diagram, vertically to change the PI controller gain Kp or horizontally to change the integral time Ti=Kp/Ki, or the PI parameters are changed directly with the sliders, and all the graphics are updated continuously.

Fig. 5. Remote experimental system in closed-loop with the PI controller designed in the previous step. The Set Point and the trace scale can be adjusted to show if experience confirms the theory.

5.1 Integration in Learning Platforms

A good integration of remote lab experiments and simulation tools in existing learning platforms is important to give students an efficient learning experience. The software should take care of aspects which have no educative value, such as spending time learning a user interface which is used only once, or entering lengthy data at the keyboard.

But the flow of information which is a key aspect of the methodology should remain clear to the students. The best approach depends not only on the topic, but also on the students and on the teaching context with respect to other courses.

At University of Applied Sciences and Arts Western Switzerland, these interactive experimentation and simulation tools are integrated with Moodle which also offers video clips where teachers explain the theory and what is expected from the students, and static PDF files. Another approach, more linear, is also possible. With MOOC platforms such as edX, the matter is divided into short sections; sessions with lab experiments the computation tools is more focused, and the intermediate results which must exchanged are managed by the platform [2].

Thanks to HTTPS support, the interactive lab experimentation and simulation tools for remote laboratories detailed in this paper can be integrated seamlessly in the Graasp platform introduced in [3] to support online learning activities. The graasp.eu digital education platform [4] has been developed with the support of the European Commission and Swiss public funds for enabling online personal, collaborative and inquiry learning. It is open to any educational institution or nonprofit organization promoting active learning and knowledge sharing. The platform has also been adapted to be used in under-connected areas [5].

Graasp has been extensively used during the COVID19 crisis not only by engineering students for blended design and computational thinking activities [6], but also by school teachers to share content with parents and children carrying out school activities at home during the confinement period. The simplicity to create activities integrating multimedia resources and the possibility to share them with students just with a secret link were the main reasons for this swift adoption.

5.2 Client-Server Architecture

For anyone observing the evolution of technologies involved in remote experiments since Internet access became more and more accessible in the nineties, it is quite obvious the environment has changed very significantly. On the client side, mostly for security reasons, Java applets, Flash Player applications, and other native browser plugins have been replaced by JavaScript web applications. The performance which can be obtained in JavaScript has increased dramatically thanks to just-in time compilers and the competition between the main browser editors. Hybrid architectures where heavy computations are performed on the server and where the client is limited to user input and result visualization, such as AJAX or Jupyter, can often be avoided. This has the benefit to reduce the cost of the server infrastructure, security and maintenance.

Communication protocols are still open and give the freedom to develop innovative solutions. Learning platforms are not standardized to a level which would permit an easy migration should it be required. This is why for this remote experimentation setup, effort has been put mostly on the remote experiment and client-side tools with data under the direct control of the client rather than based on cloud storage.

5.3 Security Aspects

The connection between the student computers (clients) and the servers connected to the lab experiments must be secured against malevolent actors. The lab experiments and server computers must be protected against unauthorized connection attempts which could steal resources, impair their usage by the students they are destined to, and possibly damage them by submitting repeated extreme commands. The student computers should be protected against malware, phishing attempts, or any content which would tarnish the lab institution. And the experimental data integrity should be guaranteed to ensure students can rely on their remote experiment outcome.

On the remote lab side, servers are connected to wired networks which are difficult to attack. The weakest side is the client computers and their Internet access: Wi-fi connections have become the norm on laptops and tablets, and the security of public wi-fi access points is often low or impossible to assess. To improve overall security, the use of TLS (SSL) encryption has become prevalent (HTTPS URLs), with more and more restrictions enforced to discourage practices known to be easily abused. Modern browsers flag pages served without encryption as unsecure, refuse unencrypted auxiliary resources accessed from encrypted pages ("mixed content"), and remember if a site has been served via HTTPS to not switch back to unencrypted HTTP in future accesses ("HTTP Strict-Transport-Security"). Cross-origin constraints restrict possible implementations further (cross-origin refers to web content in the same page obtained from different servers).

In addition to encryption, TLS guarantees the server identity: it prevents a malevolent third-party to impersonate a server, collect sensitive data sent by the client and provide them with fake replies. Identity is based on certificates with a chain of trust: browser (or the operating system) has a list of trusted parties, and thanks to public key signatures, a chain of certificates leads to the server certificate linked to its IP address. Currently, this restricts servers to fixed IP addresses.

These constraints are expected to continue tightening. For instance the DNS protocol (domain name system) used to obtain the numerical IP address from the domain name, which is still used without encryption and can be easily diverted (sometimes for legitimate reasons), is replaced by DNS-over-HTTPS in the main browsers, which can lead to other concerns. Applications such as remote lab experimentation are bound to follow closely these trends.

The interactive tools are served from a server implemented in LabVIEW to the students' browsers via HTTP 1.1 (a simple request/reply protocol where the connection is closed after one or a few requests) and WebSockets (a lasting bidirectional stream of messages starting with an HTTP handshake). Both of them are supported natively by browsers. For the reasons described above, the encrypted versions of these protocols is used, HTTPS and WSS. On WebSocket, messages are encoded in JSON, a good compromise between legibility and conciseness which is supported natively in browsers and elsewhere. Control parameters and measurement data are sent as numbers, and the video stream is sent as JPEG images in the same connection, the easiest way to ensure synchronization between video and real-time signal traces. The video quality suits well its purpose, a direct way to assess the system functioning.

Web browsers are de-facto standard environment for client applications. Client applications written in JavaScript enable students to fully interact with remote systems, through real-time video streaming, oscilloscope or other data display systems as well as control panel to specify the controller parameters. Thanks to the Smart Device (SD) API, different client applications can connect to different SD services through the API standardized in the framework of the IEEE P1876 working group [7]. As for Communication protocol, WebSocket are currently the most effective way of exchanging information between the client web application and the remote server.

6 SUDAC Experimental System

Along with the existing pool of educational remote laboratories available at University of Applied Sciences and Arts Western Switzerland and at Swiss Federal Institute of Technology, the project aims at offering a new pool of sustainable renewable energy installations operating under real weather conditions to be installed in the southern countries (Iran, Lebanon, Niger and Djibouti). The Swiss students in engineering disciplines are supposed to use the photovoltaic cells installed in the south countries to do different real measurements (angle, temperature, PV voltage output, electrical power, ...) and to test control algorithms under warmer and more severe weather conditions than those typically observed in Switzerland. The proposed system (Fig. 6) offers the ability to remotely conduct experiments with real biaxial photovoltaic cells. The biaxial solar tracking system actuated by a motorized structure allows adjusting the azimuth and elevation angles to obtain the Maximum Power Point Tracking (MPPT). A camera offers continuous live video streaming of the test site in addition to other remote measurements. A solar charge controller and a battery complete the set of PV cells experimentation system.

Fig. 6. SUDAC experimental system.

7 Conclusions

The results of this study point to important consensus in the affordances of both remote labs and simulations tools for science learning and critical elements of the user interface that enhance student engagement across both types of tools. An appropriate combination of lab experimentation and simulation tools offers the students the opportunity to conduct a real experiment while using the lab-based simulation tools for experiment modeling, controller designing, analyzing and interpreting data.

Acknowledgment

The work presented in this paper is carried out in the framework of the MOOLs project co-funded by the Swissuniversities Development and Cooperation network (SUDAC) program. The program seeks targeted support for collaboration between the various types of Swiss higher education institutions and their partners from the Global South in order to achieve an excellent standard of education and research on global challenges.

References

1. H. Saliah-Hassane, M. Castro Gil, D. Gillet, M.L. Petrie, J.B. Da Silva, L.P. Zapata Rivera, L. Mellos Carlos, J. W. Shockley, J. Zalewski, G.R. Alves, E. Sancristobal Ruiz, "Special Session—Online Laboratories in Engineering Education: Innovation, Disruption, and Future Potential", Proceedings of the IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), pp. 1228-1232, 2019.
2. C. Salzmann, Y. Piguet, and D. Gillet, "New Tools for MOOC/MOOL to Sustain Continuity of Experimentation in Control, 12th IFAC Symposium on Advances in Control Education (ACE), Philadelphia, 2019, pp. 254-259.
3. F. Moghaddam, A. Vaccari, T. Moniquet, Ch. Salzmann, Y. Piguet, S. Saab, R. Jaafar, A. Abolfazl Suratgar, Q. Shafiee, N. Dan Lamso, H. Alzouma Mayaki, H. Ali Barkad, H. Saliah Hassane, and D. Gillet, "Massive Open Online Labs (MOOLs): An Innovative Solution to Achieving SDGs in the Global South", Proceedings of 5th Experiment International Conference (exp. at'19), pp. 394-398, 2019.
4. M.J. Rodrguez-Triana, S. Govaerts, W. Halimi, A. Holzer, Ch. Salzmann, A. Vozniuk, and D. Gillet, "Rich open educational resources for personal and inquiry learning: Agile creation, sharing and reuse in educational social media platforms", International Conference on Web and Open Access to Learning (ICWOAL), Dubai, November 25, 2014.
5. D. Gillet, B. Tijani, S. Beheton, J.C. Farah, D. Dikke, A. Noutahi, R. Doran, N. RC Gomes, S. Rich, T. de Jong, C. Gavaud, "Promoting and Implementing Digital STEM Education at Secondary Schools in Africa", Proceedings of 2019 IEEE Global Engineering Education Conference (EDUCON), pp. 698-705, 2019.
6. J.C. Farah, A. Moro, K. Bergram, A. Kumar Purohit, D. Gillet, and A. Holzer, "Bringing Computational Thinking to non-STEM Undergraduates through an Integrated Notebook Application", The 15th annual European Conference on Technology-Enhanced Learning (EC-TEL), Heidelberg, Germany, 14-18 September 2020.
7. 1876-2019 - IEEE Standard for Networked Smart Learning Objects for Online Laboratories, IEEE Education Society/Standards Committee (EDU/SC), 2019, https://standards.ieee.org/standard/1876-2019.html