

Developing E-Course Robust Constrained PID Control

[doi:10.3991/ijet.v6iS1.1617](https://doi.org/10.3991/ijet.v6iS1.1617)

M. Huba^{1,2}, P. Ľapák¹

¹ Slovak University of Technology in Bratislava, Bratislava, Slovakia

² FernUniversität in Hagen, Hagen, Germany

Abstract—This paper informs about the latest development of the blended learning course Robust Constrained PID Control and describes in details problems faced in evaluating independent work of students with real time experiments that represent the main element of the applied “learning by doing” and “learning by experimenting” approach. Basic features of the developed system that evaluates consistency of student’s results with the measured experimental data and possibilities of further system expansion to automation of offering to student constructive feedback are discussed. Simultaneously, by comparing all data of the Matlab workspace the developed system serves although as an excellent anti-plagiarism tool.

Index Terms—anti-plagiarism, real experiments, automated assignment marking, Matlab/Simulink, robust control

I. INTRODUCTION

For more than two decades, at MSc level, the “Constrained PID Control” course has been developed and piloted [1]. Its aim was to integrate, explain, verify, compare and evaluate different traditional and new PID control structures based on linear models with dominant dynamics up to the 2nd order with possible long dead time. This all was taking into account also constraints given on the manipulated variable, or the state variables. Despite apparent simplicity of the PID control, from the beginning we were faced with the well known problem (see e.g. [2]) that already in linear context it is not easy to find good controller settings, especially from the point of view of robust control. The education is difficult especially when dealing with time delay systems, where different interactive tools [3, 4] are used to compensate shortage of reliable and easy to use control theory. In order to enable to students to cope with the peculiar problem of reliable robust controller tuning, a group of teachers from STU (Slovak University of Technology) have brought into the education a high number of low-cost plant models including own micro-processor for the data processing and communication. These are able to be connected to standard computers via the USB port. This solution does not require purchasing special converter cards and simplifies the overall control communication that enables to carry out real time control tasks in any standard computer room, or even by own computers at home.

From the huge amount of experiments carried out by our students it was obvious that in order to guarantee required control quality over total operating range of the inherently nonlinear and time variable plants the traditional nominal controller tuning should be replaced by

robust tuning procedures. The basic problem was given by the fact that the available methods do cover in an effective way neither the robust and constrained aspects of the control design.

The situation started to change at the beginning of 2009, when new principles for a simple and easy to understand and use robust constrained controller tuning were discovered [5-7]. These are now being integrated into the older course structure, whereby many new and surprising results are achieved. The course students are participating in the re-evaluation of the older controller design and tuning method and in piloting and testing the newly developed robust control methods.

II. COURSE STRUCTURE

The course uses the learning environment of the LMS Moodle, where the study materials, assignments and the test are available. So the students are supported by:

- the printed textbook [8] with many examples and exercises, summaries and reflection points,
- permanently updated electronic course materials (available from the LMS Moodle) containing
 - Matlab/Simulink programmes for simulation, robust design and control experiments,
 - FLASH animations of basic problems,
 - e-books,
 - computer tests for self-evaluation, etc.,
- physical models of real plants to be controlled (Fig.1) that are either sent or given to each student or made accessible via Internet,
- face-to-face workshops (if required by students) and
- tutors.

Up to this moment, advantages of the plant models developed for our course were discovered also by 10 other university partners in Slovakia, Germany, Norway, Slovenia and India. Growing interest in this field can e.g. be illustrated by the papers presented at the Int. Conference Process Control in June 2009 [9].

Now, the conditions to widen spectrum of our products and services by the total course delivery and management ranging from the plant models, through the course materials up to the Moodle learning environment, are gathered. New automated authenticity test for the experimental results is being developed based on the measured data analysis in Matlab. The automated assignment marking was introduced in several lectures as well.

III. REAL EXPERIMENTS

The “Robust Constrained PID Control” course represents the set of case studies devoted to constrained time-delayed PID control. Student can compare and verify the recommended solutions as well as many other well-known ones proposed by different authors and to construct his/her own opinion. As the course is supported by various types of materials typical for modern education including simulation tools and real plants, course participants can draw their own conclusions based on their own experience. In this constructive way – by solving real problems in a quasi-authentic environment and collecting personal experience – the transfer of the research results into practice is promoted and the gap between the theory and practice, characteristic for the present control theory, is well on the way to being decreased.

To enable each student to work actively and individually in a quasi-authentic framework with real time experiments, we need to have numerous plant models that guarantee:

- clear physical “visibility” of the controlled dynamics,
- time constants in the range ms-few minutes,
- safety manipulation,
- reasonable price,
- availability of sensors and actuators,
- easy maintenance,
- connectivity to standard computers without special converter cards,
- possible approach via Internet,
- plants with different degree of “the control difficulty”,
- plants offering broader spectrum of dynamics to be
- controlled.

As a typical representative of a plant having all required features we can mention the thermo-optical plant (Fig.1). This product of several decades of development offers measurement of eight process variables (controlled temperature and its filtered value, ambient temperature, controlled light intensity, its filtered value and its derivative, the ventilator speed of rotation and its motor current). The temperature and the light intensity control channels are interconnected by 3 manipulated variables: the bulb voltage (the heat & light source), the light-diode voltage (the light source) and the ventilator voltage (the system cooling). The plant can be easily connected to standard computers via USB, when it enables to work with the sampling periods 40-50 ms and larger. Within the Matlab/Simulink scheme the plant is represented as a single block, limiting use of costly and complicated software package for the real time control. So, the usual process-computer communication based on standard converter cards (that is also supported) is necessary just for more demanding applications requiring higher sampling frequencies.

IV. THE ANTI-PLAGIARISM SYSTEM

The disadvantage of the real experiment is, there are too many aspects of the design to which the students should receive a feedback and which should be considered as bases for marking his/her work. Some important features



Figure 1. Thermal-optical plant

of his/her work cannot be even seen from the written report, in other words, it is not possible to mark student’s work using only the measurement trends in the report. The solution we suggest is to analyze the data the students measured. It brings a bit more work for students to upload the data together with the report; however, it saves a lot of time for the teacher. As the matter of fact it also saves the students’ time as well, because they usually do have to use the data from the previous assignments for their future work.

The file format for the measured data chosen is the Matlab workspace file. The reason was that the experiments were made in Matlab/Simulink environment and the Matlab itself is great software for creating such a tool in.

The students upload one zip-file containing the report and several workspaces with the required data. The workspaces must have specified names and the variables inside the workspace must have specified names as well.

At first, the file was unzip into a temporary folder. Then, the folder is searched for the required files. Each time the workspace file is loaded the existence of the required variables is checked to prevent errors during the script execution. However, if some variables are missing, the script tries to work without them. Typical example is when student has to save the output of the system as the variable y , the control signal as u , the setpoint as w , and the time vector as t , students tend to save the time vector as $tout$ which is the Simulink’s default variable name for the simulation time. They tend to save the output as the variable $y1$ instead of y etc. The script searches for a few usual variables as well. After these operations, the script stores the student results in its own variables for later comparison.

The course is based on real experimenting. To explain what is analyzed in the scripts, typical lesson are used to describe it.

The assignment always contains a short guide. The typical lesson starts with an identification of the plant. The students try to find the transfer function of the system as the first step. They have to measure e.g. the step response (Fig. 2) or the input output characteristics of the real plant.

The first thing to analyze is the sample time students used in measuring the step response. The default value of the sample time in the Simulink blocks used for communicating with the plant is 1s. However when using the light channel the 10 times faster sampling has to be used to obtain relevant results. Then the script makes the identification itself and compares the results with the students’ ones.

After system identification, appropriate controller settings have to be computed. The controller structure and the formulas to compute its parameters are to find in the guide to the assignment (Fig. 3).

There is more to analyze for the script here than in the first case.

The script computes the parameters two times: once the controller parameters are based on the automatic identification, second time the script computes the controller parameters but it uses the identification made by student. The purpose is to check if the student did compute the parameters using his own data. The amount of the points student earns depends on the deviation in the results.

After designing the controller students use simulation to verify the controllers design.

The simulation is made by script as well, using the students' controller data from workspace. If the simulation made by script and the simulation stored in the workspace file uploaded by student are not the same the student has made a mistake in the model, or the student did not use the data obtained by the identification.

After successful simulation the lesson continues with the real-time experiment (Fig. 4).

The simulation and the real-time experiment results are compared then. The real experiment results have to be similar to the simulation results, however there are some differences. These differences and overall controller's performance have to be summarized and commented by student.

In this step the highest number of elements is being analyzed by the script. The results are compared with the simulation results. If the simulation results and the real experiment results are too much different, there is a high probability that the real experiment was not made using the student's data. Again, the student may earn some points even in the case when the controller parameters are wrong, e.g. for making the Simulink model correctly.

The quality of control is analyzed then. The controllers designed in this course should guarantee transients with no overshooting or with a specified tolerated overshooting. Therefore, the maximal overshooting achieved is analyzed. The most frequently used Integral of Squared Error (ISE) and Integral of Absolute Error (IAE) performance measures are taken into account. The steady state error check is in the script as well. The presence of steady state error or the bigger overshooting than defined is the kind of error for which the student loses all the points for the experiment. The student has to know that something is wrong if there is big overshooting or the steady state error.

Finally all the students' results are compared, if there is the match between two students' results, these are marked as plagiary.

V. CONCLUSION

Broad use of plant models in our course has created a lot of new problems. We have many students producing a high number of reports to experiments that need to be marked and completed by relevant feedback to students. The measurement data collected up to now from students will help us to develop much better tool for automated marking in future. It will help us to determine range of the results which are correct for a particular plant, or the range of results which are obviously wrong. Before the course

started we were not able to make the marking of the assignments automatically due to the lack of information on the plants. Now we can use the average values from the assignments to set up the marking rules correctly. This approach can be used online in future. The student would upload the assignment e.g. to MatlabServer which would provide immediate feedback with some tips how to improve the assignment.

ACKNOWLEDGMENT

This work was partially supported by Project VEGA 1/0656/09: Integration and development of nonlinear and robust control methods and their application in controlling flying vehicles and project KEGA 3/7245/09 Building virtual and remote experiments for network of online laboratories.

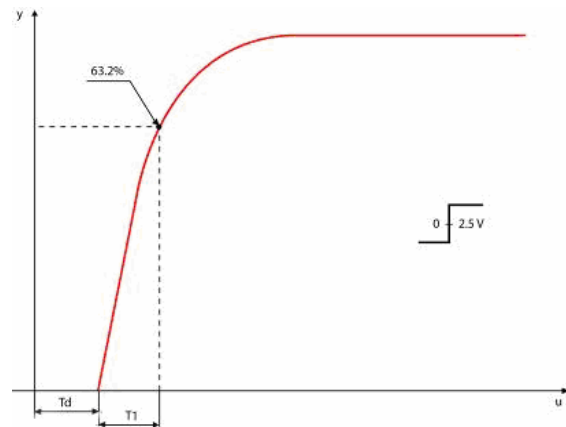


Figure 2. Typical identification of the simplest dynamical processes based on the Average Residence Time – sum of the dead time and single time constant

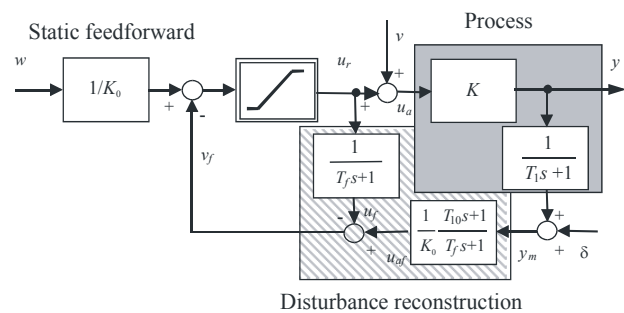


Figure 3. PI_0 -controller

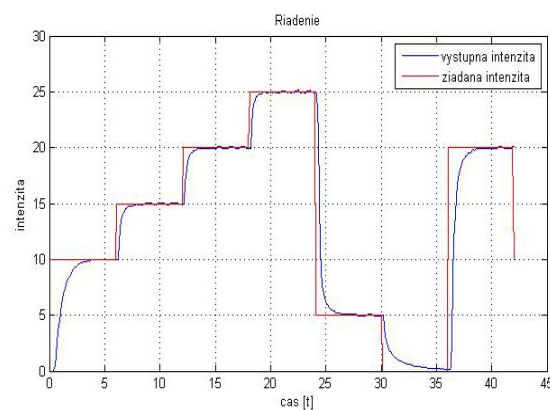


Figure 4. Real-time experiment results

REFERENCES

- [1] Huba, M., Šimunek, M.: Modular Approach to Teaching PID Control. In: IEEE Transactions on Industrial Electronics, Vol.54 No.6 December 2007, pp.3112-3121. [doi:10.1109/TIE.2007.907024](https://doi.org/10.1109/TIE.2007.907024)
- [2] Datta, A., Ho, M.T. and S.P. Bhattacharyya: Structure and synthesis of PID controllers. Springer, London 2000.
- [3] Guzmán, J.L., et al.: Interactive tool for analysis of time - delay systems with dead - time compensators. Control Eng. Practice, Vol. 16, 7, 2008, 824-835
- [4] Normey-Rico, J.E. et al.: An unified approach for DTC design using interactive tools. Control Eng. Practice 17, 2009, 1234-1244. [doi:10.1016/j.conengprac.2009.05.007](https://doi.org/10.1016/j.conengprac.2009.05.007)
- [5] Huba, M.: Robust Design of Integrating Controllers for IPDT Plant. Edits: Fikar, M., Kvašnica, M., In Proc. 17th Int. Conf. Process Control '09, Š. Pleso, 2009, 353-357, <http://www.kirp.chtf.stuba.sk/pc09/data/papers/109.pdf>.
- [6] Huba, M., Marko, L., Bahnik, P., Oravec, I.: Numerical Issues in Designing PI Controller for IPDT Plant. *ibid*, 57-64, /110.pdf.
- [7] Huba, M.: Designing Robust Controller Tuning for Dead Time Systems. IFAC Int. Conf. System Structure and Control 2010, Ancona, Italy
- [8] Huba, M.: Theory of Automatic Control 3: Constrained PID Control. STU Bratislava 2006 (in Slovak).
- [9] Papers dealing with the thermo-optical plant control presented at the 17th Int. Conf. Process Control '09, Š. Pleso, 2009
<http://www.kirp.chtf.stuba.sk/pc09/data/abstracts/023.html>
<http://www.kirp.chtf.stuba.sk/pc09/data/papers/082.pdf>
<http://www.kirp.chtf.stuba.sk/pc09/data/abstracts/092.html>
<http://www.kirp.chtf.stuba.sk/pc09/data/abstracts/111.html>
<http://www.kirp.chtf.stuba.sk/pc09/data/papers/069.pdf>

AUTHORS

M. Huba received the MSc. and PhD. Degrees in Technical Cybernetics from Slovak University of Technology in Bratislava in 1974 and 1982, respectively. Since 1989 he was a Senior Lecture and Head of the Control Theory Group of the Institute of Control and Industrial Informatics and from 2008 Full Professor at the Faculty of Electrical Engineering and Information Technology. In 1996 he initiated establishment of the university Distance Education Centre and worked as its director up to 2008. Since 2008 he was also active in research at the FernUniversität Hagen, Germany, where he now continues as the Mercator visiting professor. He is author and co-author of more than 200 papers in journals and proceedings of international conferences and 20 monographs on Constrained, Nonlinear and Remote Control and about e-Learning (e-mail: mikulas.huba@stuba.sk).

P. Ťapák is with the Institute of Control and Industrial Informatics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava, Slovakia, (e-mail: peter.tapak@stuba.sk).

This article was modified from a presentation and paper at The 10th International Virtual University Conference, in 10-11 December 2009, Bratislava, Slovak Republic.

Received March 29th, 2011. Published as resubmitted by the authors April 20th, 2011.