DESIGNING PID CONTROLLERS USING MATLAB-SIMULINK VIA THE INTERNET

Ľ. Čirka, M. Kalúz, J. Oravec, N. Míková

Institute of Information Engineering, Automation, and Mathematic
Faculty of Chemical and Food Technology, STU in Bratislava
Radlinského 9, 812 37 Bratislava

Abstract

In this paper, we present an Internet version of software for PID controller tuning, which is the open Web-based alternative to our former toolbox for control design. The software represents a user friendly tool for simple step-response-based identification of a process model, fast PID controller tuning, and effective investigation of control performance.

1 Introduction

The aim of this paper is to present an Internet version of software for PID controller tuning called MWS PIDDESIGN, which is the open Web-based alternative to our former toolbox PIDDESIGN for control design [1].

PID controllers are commonly used in industry and they still represent the major portion of all low-level controllers used in practice [2]. The biggest advantage of PID controllers is their low complexity, robustness and possibility to realize them also in digital way. MWS PIDDESIGN is oriented mainly on PID controller tuning. The application enables to tune PID controllers using various analytical and experimental methods, see e.g. [2 - 8].

Numerous commercial and free tools for PID tuning are currently available on the market and in open source community (see e.g. [9 - 11]). An example of an interesting on-line tool for interactive design of PID controllers is the PID Control Laboratory [12]. It allows to optimize the controller's parameters according to desired requirements. The disadvantage of this tool is that is requires the Java Runtime Environment to be run.

The MWS PIDDESIGN is intended to be used mostly as the supplementary tool for control education. It can be useful also for those users, who are looking for the tool that easily handles the model identification from step responses, filtering of noisy data and featureful design of PID controllers. From the user point of view, the MWS PIDDESIGN does not require any specialized software and it can be run in common Web browser.

2 PIDDESIGN toolbox

Internet version of software for PID controller tuning is designed on the basis of the PIDDESIGN toolbox. The toolbox PIDDESIGN has been developed at the Institute of Information Engineering, Automation, and Mathematics at the FCFT STU in Bratislava [1]. The toolbox uses MATLAB graphic user interface to determine the PIDDESIGN as visual and user friendly tool (Fig. 1). PIDDESIGN represents useful tool for:

- a simple identification of a controlled process,
- the fast PID controller tuning using various methods,
- an effective evaluation of control performance using simulation of control.

All of these tasks can be solved independently. If a transfer function of the controlled process is unknown, the software enables to identify the controlled process from its step response. The identified process model can be described as the transfer function:

$$G(s) = \frac{K}{(Ts+1)^n} e^{-Ds} \qquad (1)$$

for the aperiodic system or

$$G(s) = \frac{K}{T^2 s^2 + 2\xi T s + 1} e^{-Ds} \qquad (2)$$

for damped periodic step response where n is the order of the system, K is the gain, T represents the time constant, ξ is the damping coefficient, D represents the time delay.

It is possible to design P, PI, PID or PD controllers. Control performance obtained using various tuned controllers can be simply compared using simulation of control in different conditions of input constraints, various values of set-point, disturbance, integral windup elimination parameter, and in presence of the uncertainties in the controlled model.

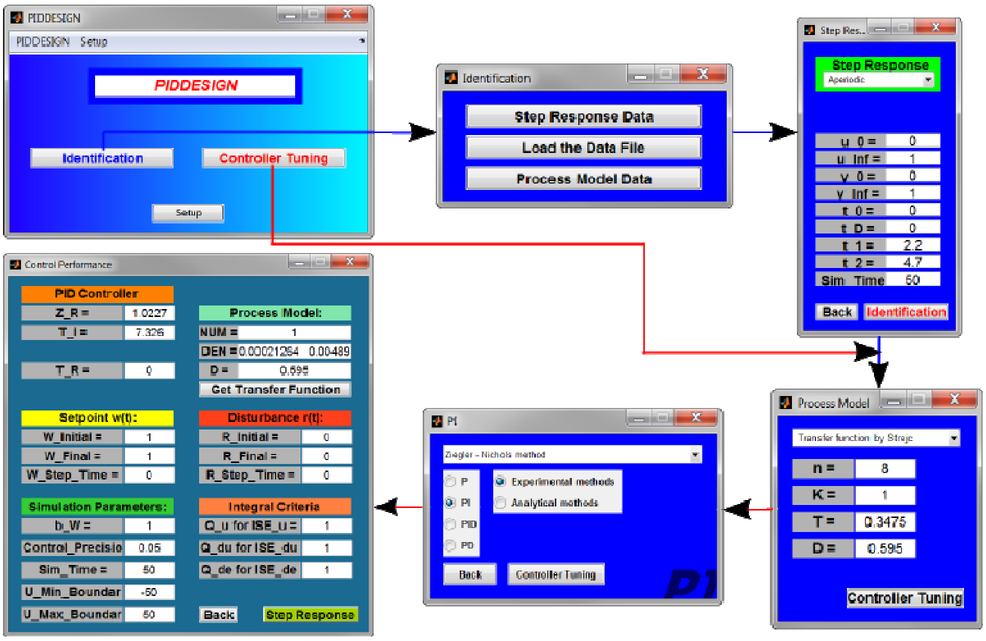

Figure 1: PIDDESIGN toolbox diagram

3 Web Server

Our approach is based on HTTP Socket Server that is based on independent open-source project Web Server (WS), developed and published by Dirk-Jan Kroon [13]. This server executes native MATLAB code over the Web. The server side service is emulated in by Java package classes *java.net.** which are designed for I/O network communications. The service in set up to process common HTTP requests for GET and POST methods sent from client side Web pages. The source code of the server consists of M-file executable scripts and functions containing base methods for handling HTTP request/response model and methods for local execution of server side processing scripts.

Standard application that uses WS consists of three parts:

1. **Input template** – input form is a simple HTML document. It contains elements (arrays) used to get input data for a computational m-file.

2. **m-file** – MATLAB code for checking the form input data. In the next step, data are sent to a Simulink scheme that is running. Finally, resulting simulation data are sent back to HTML output document in an output structure. Each m-file tests all input parameters first, their data types and dimensions if needed. To prevent errors when processing m-file, JavaScript form validation is applied before the data are sent to WS. This helps to avoid the situations when user generates the data that produces error due to the wrong format or unfeasibility of numerical solution.

3. **Output template** – simple HTML output document that is displayed as the result in the web browser.

4 MWS PIDDESIGN application

MWS PIDDESIGN is an Internet application of the PIDDESIGN toolbox. Application is divided into two independent parts: Identification and Controller tuning (Fig. 2).

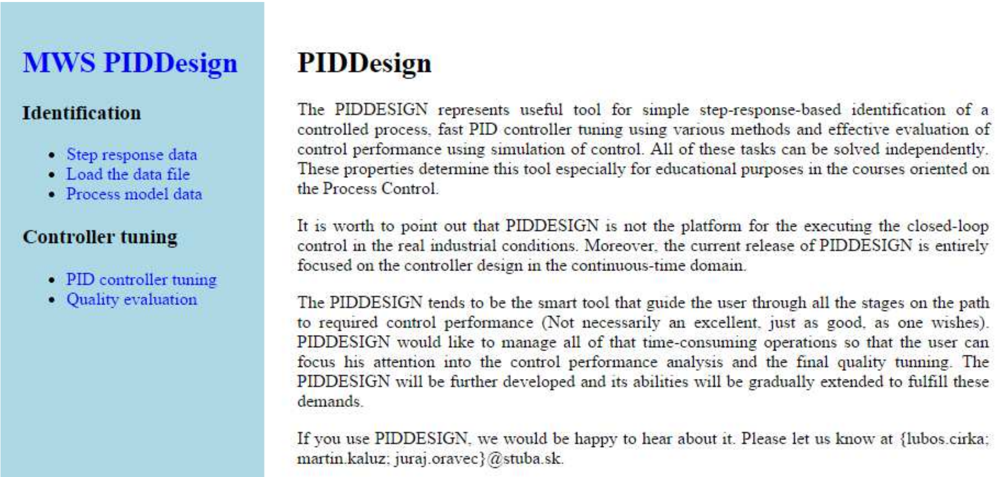

Figure 2: MWS PIDDESIGN home page

First part offers three identification possibilities:

1. **Step response data** – enables to identify the controlled process directly from data obtained from the measured step response (Fig. 3);

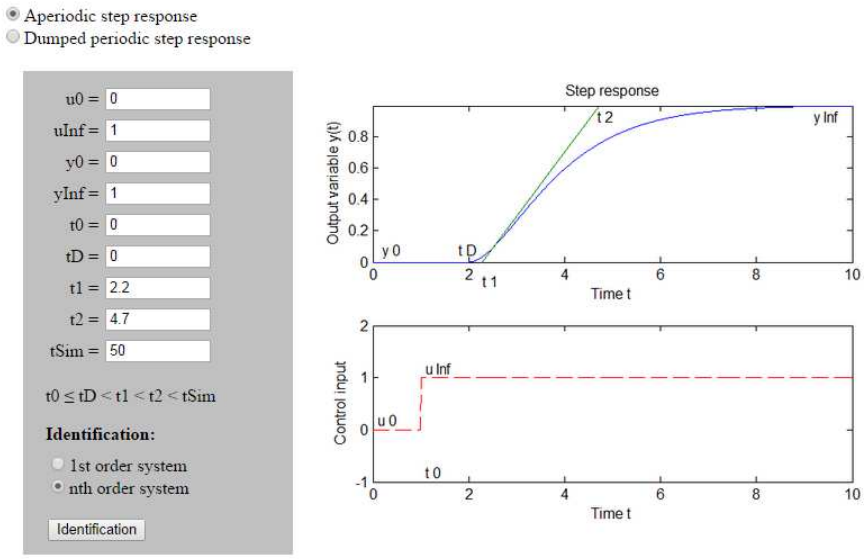

Figure 3: Identification: Step response data of aperiodic system

2. **Load the data file** – user cannot directly upload the data file containing recorded step-response data. However, it can insert data from a data file into text area (Fig. 4). The considered structure of the data is as follows, the first column vector represents a time, the second column vector of measured values of output variable, and the third column vector of values of manipulated variable. If loaded data are noisy, the user can use the filtration before identification;

3. **Process model data** – enables to identify process model with required properties of transfer function. The parameters of the model described by the (1) or (2) or general transfer function (3) can be simply set (Fig. 5).

$$G(s) = \frac{N(s)}{D(s)} e^{-Ds} \tag{3}$$

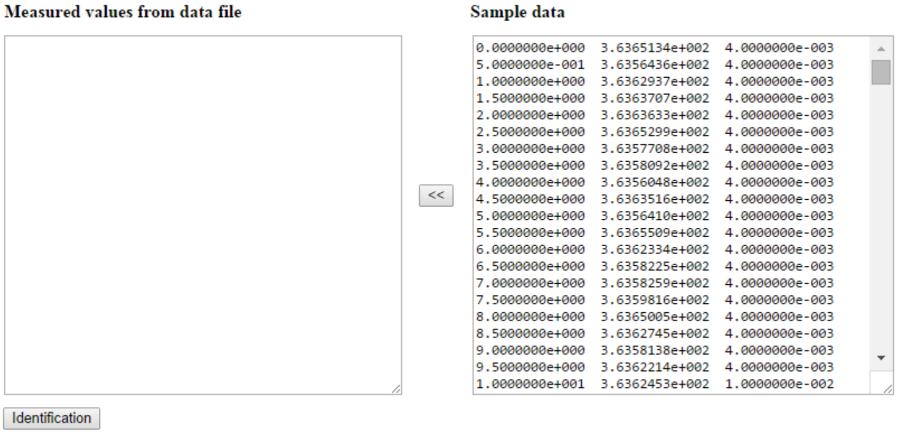

Figure 4: Identification: Data loaded from file

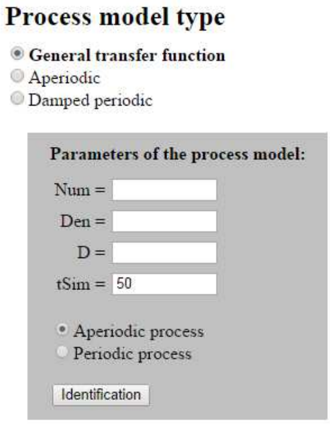

Figure 5: Identification: Process model data

The second functional part of MWS PIDDESIGN deals with the PID controller tuning (Fig. 6). PID controllers can be designed for controlled process models with either dumped periodic or aperiodic step responses described by the transfer functions (1) or (2). It is possible to choose a P, PI, PD or PID controller and a type of a tuning method. For simpler handling, the methods for controller tuning are divided into two main groups: analytical and experimental methods. Various types of analytical and experimental methods can be used for controller tuning.

Calculated parameters of the tuned controller are shown in a new page (Fig. 7). Some additional parameters are displayed as well. These are: parameters of the transfer function; simulation setting for set-point and disturbance signals; and values of integral criteria of quality. These parameters can be modified and the tuned controller can be so tested in the presence of model uncertainty. The properties of the closed loop with the tuned controller can be analysed by simulation of control. The result of quality evaluation of control is shown in Fig. 8.

More details on the identification and controller tuning can be found in [1].

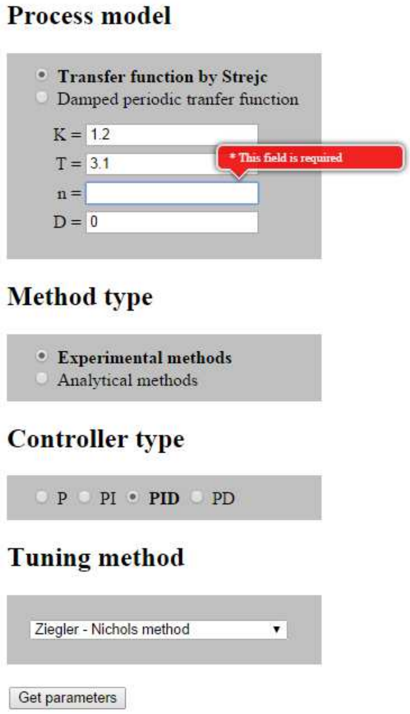

Figure 6: Controller tuning (with sample validation form)

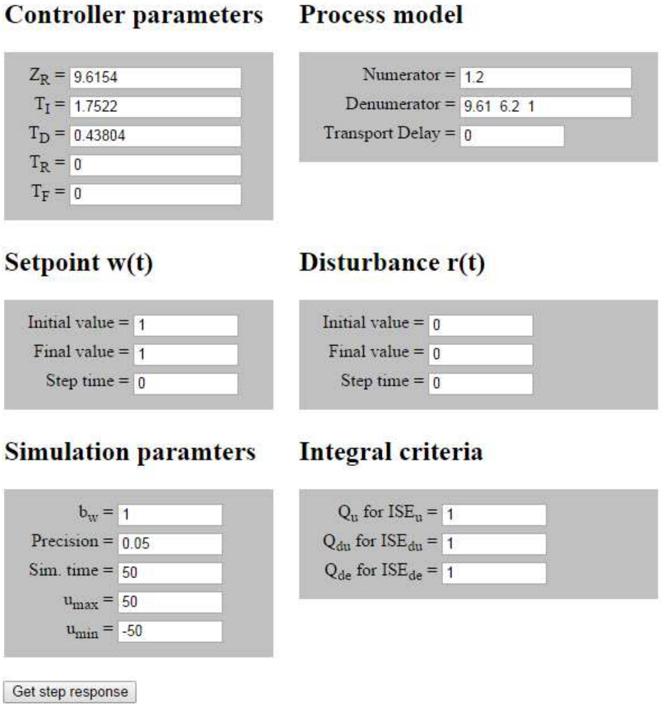

Figure 7: Controller, process model, and simulation parameters

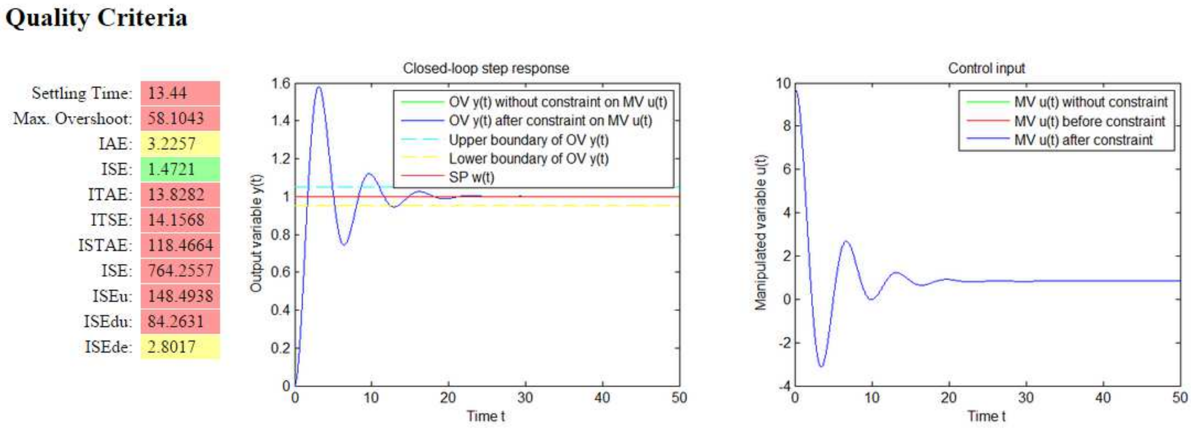

Figure 8: Quality evaluation of control

5 Conclusions

Compared to the original PIDDESIGN toolbox [1], the Internet version MWS PIDDESIGN make can be used from any place with Internet and without any software dependences. The MWS PIDDESIGN application is an open system. In the future, we plan to extend our application by new features (identification of stochastic systems, cascade control, etc.).

Both versions (toolbox and Internet version) are freely available for academic research. The MWS PIDDESIGN homepage can be found at *http://www.kirp.chtf.stuba.sk/~cirka/mwspiddesign/*.

References

[1] J. Oravec and M. Bakošová. PIDDESIGN – Software for PID Control Education. In IFAC Conference on Advances in PID Control, Brescia, Italy, 2012.

[2] K. J. Åström and T. Hägglund. The future of PID control. *Control Engineering Practice*, 9:1163–1175, 2001.

[3] D. Xue and Y. Q. Chen. *Analysis and Design with MATLAB (Advanced in Design and Control)*. Society for Industrial Mathematics, 1st ed. edition, 2008.

[4] M. Bakošová and M. Fikar. *Process Control* (in Slovak). STU Press, Bratislava, 2008.

[5] J. Mikleš and M. Fikar. *Process Modelling, Identification, and Control*. Springer Verlag, Berlin Heidelberg, 2007.

[6] B. A. Ogunnaike andW. H. Ray. *Process dynamics, modeling and control*. Oxford University Press, New York, 1994.

[7] A. Vítečková and M. Víteček. *Automatic control fundamentals* (in Czech). TU Ostrava, 2006.

[8] K. H. Weng, C. H. Chang, and S. C. Lisheng. Tuning of PID Controllers Based on Gain and Phase Margin Specifications. *Automatica*, 31(3):497–502, 1995.

[9] Mathworks. *PIDTuner*, 2014. [Online: http://www.mathworks.com/help/control/ref/pidtuner.html]

[10] J. D. Rojas and R. Vilanova. Data-driven Robust PID Tuning Toolbox. In *IFAC Conference on Advances in PID Control*, Brescia, Italy, 2012.

[11] J. L. Guzmán, D. Rivera, M. Berenguel, and S. Dormido. i-pIDtune: An Interactive Tool for Integrated System Identification and PID Control. In *IFAC Conference on Advances in PID Control*, Brescia, Italy, 2012.

[12] M. Schlegel and M. Čech. Internet PID controller design: www.pidlab.com. *IBCE´04*, p. 1-6, ENSIEG, Grenoble, 2004.

[13] D. J. Kroon. Web server. online, url: http://www.mathworks.com/matlabcentral/fileexchange/29027, 2011.

Acknowledgements

The authors gratefully acknowledge the contribution of the Scientific Grant Agency of the Slovak Republic under the grants 1/0973/12. J. Oravec was also supported by an internal STU grant.

Ľuboš Čirka
url: http://www.kirp.chtf.stuba.sk/~cirka/
E-mail: lubos.cirka@stuba.sk

Martin Kalúz
url: http://www.kirp.chtf.stuba.sk/~kaluz/
E-mail: martin.kaluz@stuba.sk

Juraj Oravec
url: http://www.kirp.chtf.stuba.sk/~oravec/
E-mail: juraj.oravec@stuba.sk

Nikola Míková
E-mail: xmikovan@stuba.sk