Slovak University of Technology in Bratislava
Institute of Information Engineering, Automation, and Mathematics

PROCEEDINGS

17^{th} International Conference on Process Control 2009

Hotel Baník, Štrbské Pleso, Slovakia, June 9 – 12, 2009

ISBN 978-80-227-3081-5

http://www.kirp.chtf.stuba.sk/pc09

Editors: M. Fikar and M. Kvasnica

Bakošová, M., Oravec, J.: Software for PID Controller Tuning, Editors: Fikar, M., Kvasnica, M., In *Proceedings of the 17th International Conference on Process Control '09*, Štrbské Pleso, Slovakia, 524–527, 2009.

Full paper online: http://www.kirp.chtf.stuba.sk/pc09/data/abstracts/088.html

SOFTWARE FOR PID CONTROLLER TUNING

M. Bakošová, J. Oravec

*Institute of Information Engineering, Automation and Mathematics,
Faculty of Chemical and Food Technology, Slovak University of Technology,
Radlinského 9, 812 37 Bratislava, Slovakia
fax : +421 2 52496469 and e-mail: monika.bakosova@stuba.sk*

Abstract: Software for PID controller tuning PIDTOOL 1.0 was developed in the MATLAB – Simulink programming environment using its graphic user interface. The main aim of the PIDTOOL design was creating user friendly tool for simple and fast identification from data measured on the controlled process as well as for simple and fast PID controller tuning. The controlled system is identified from its step response in the form of the 1st order transfer function or in the form of the transfer function of the higher order. PID controllers are tuned using various experimental methods. The PIDTOOL properties determine its using especially for teaching purposes. The software is nowadays used for teaching at the IIEAM FCFT STU in courses oriented on process control.

Keywords: Identification, PID controller, controller tuning, graphic user interface.

1 INTRODUCTION

PID controllers belong to the most used types of controllers in the industry (Åström and Hägglund 1995) and they can be tuned by various methods.

The paper presents software for PID controller tuning PIDTOOL 1.0. It was developed in the MATLAB – Simulink programming environment and uses its graphic user interface (GUI). PIDTOOL 1.0 enables to solve two tasks, identification from step responses and PID controller tuning. Various types of experimental methods are used for controller tuning (Åström and Hägglund 1995, Ogunnaike and Ray 1994, Víteček and Vítečková 2006). Software is nowadays used for teaching purposes at the FCFT STU in Bratislava and the communication language is Slovak.

2 PIDTOOL 1.0

PIDTOOL 1.0 is software developed in the MATLAB – Simulink programming environment at the Institute of Information Engineering, Automation and Mathematics FCFT STU in Bratislava (Čemanová 2007, Oravec 2008, Oravec 2009). The main aim of the PIDTOOL design was creating user friendly tool for simple and fast identification from data measured on the controlled process as well as for simple and fast PID controller tuning especially for teaching purposes in laboratory exercises in courses oriented on process control. This is also the reason for the choice of the communication language and Slovak is used up to now. The software can be used by everybody who needs to tune PID controllers or wants to compare various types of control algorithms with simple PID control.

The software PIDTOOL uses MATLAB GUI for simpler manipulation and better visualization and MATLAB has been used for programming. Figure 1 shows the basic window of PIDTOOL 1.0.

Using the button `Identification` opens the new window (Figure 2), which offers three possibilities: to identify the system from the data which are guest from the measured and the recorded step response - the button `Step response data`; to identify the transfer function from the data file obtained by measuring of the step response – the button `Data file`; and to identify newly the system from the step response generated by the known transfer function – the button `Transfer function data.`

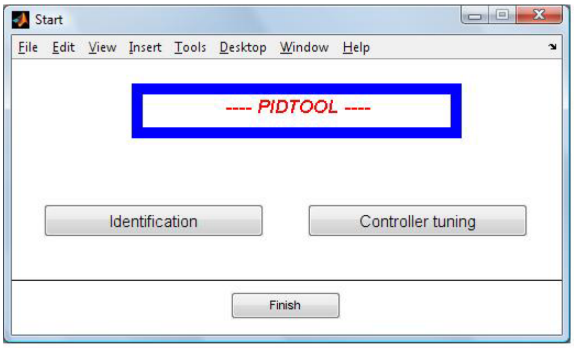

Fig. 1. Basic window of PIDTOOL 1.0

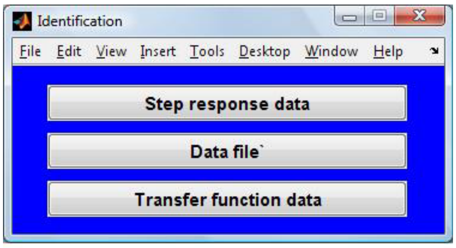

Fig. 2. Basic window of identification

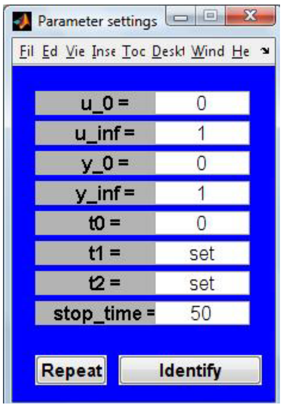

Fig. 3. Data inserting for identification

Using the button `Step response data` (see Figure 2) opens the window for data inserting (Figure 3), where following data are inserted: the initial value `u_0` and the final value `u_inf` of the step change, the initial value `y_0` and the final value `y_inf` of the step response, the time of the step change `t0`, the time `t1` of intersecting of the tangent in the inflection point to the step response with the value `y_0`, the time `t2` of intersecting of the tangent to the step response with the value `y_inf`, and the stop time of simulation of the step response of the identified transfer function. Some of these data have the default values for the sake of simple using and some have to be set. The newly set data are stored and they are used as default values in the next identification unless they are changed by the user. The button `Repeat` offers the possibility to correct inserted data and using the button `Identify` opens the new window, which is shown in Figure 4. Here, the user can choose: the identification of the transfer function of the 1st order - button `1st Order System;` or the identification of the transfer function of the higher order with the time delay - the button `n-th Order System`. The button `Back` returns the user to the previous step.

The possibility `Transfer function data` in the window `Identification` (Figure 2) is useful in the case, when the controlled process was identified in the form of the transfer function of the high order and the controller tuning needs the transfer function of the low order with the time delay. This situation can occur e.g. in controller tuning for the Smith predictor, when the time-delay process is controlled.

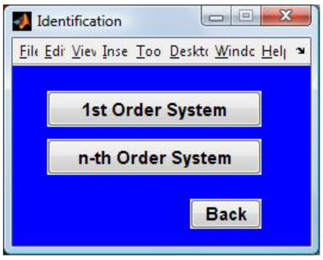

Fig. 4. Choice of the identified transfer function

The Strejc method (Bakošová and Fikar 2008, Mikleš and Fikar 2007) is used for the identification, and the transfer function

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

is identified, where n is the order of the system, K is the gain, T is the time constant and D is the time delay.

The results of identification are shown in a new window (Figure 5) and the step response obtained using the identified transfer function is displayed at the same time. The button `Compare` gives the possibility to compare the step responses of the original and the identified system in the case, when the identification from the data file or from the transfer function data was done. The button `Controller tuning` opens the window for the choice of the controller type (Figure 6).

If the transfer function of the controlled process is known, then the step `Identification` can be omitted, and the button `Controller tuning` in the main window of the program (Figure 1) can be directly used. It opens the window, where the parameters of the transfer function (1) of the controlled process have to be set. Then, the window for the choice of the controller type (Figure 6) is opened. The button `Back` gives the possibility for returning to the identification. PIDTOOL 1.0 enables to tune

P, PI, PD or PID controllers. The parameters of the controller are calculated after the choice of the type of the controller and the method for controller tuning. Various types of experimental methods are used for controller tuning (Figure 7). The calculated parameters of the chosen controller are shown in the new window (Figure 8). Here, `Z_R` is the gain and `T_I` is the reset time of the controller. If the PID controller is tuned, three parameters are shown.

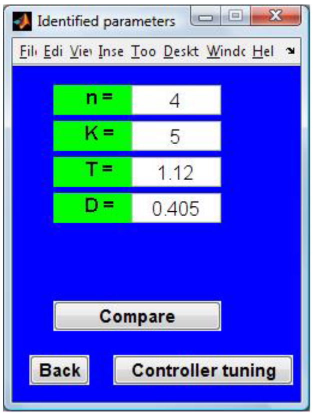

Fig. 5. Results of identification

Fig. 6. Choice of controller type

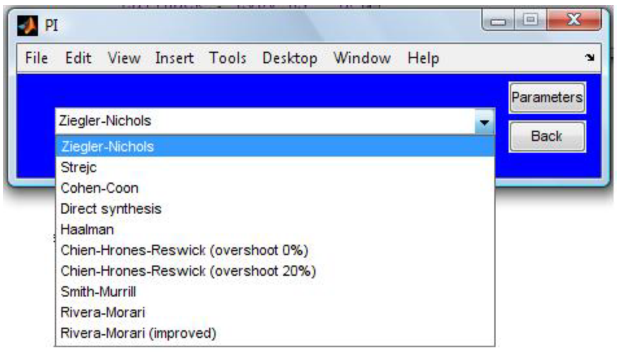

Fig. 7. Choice of method for controller tuning

Fig. 8. Designed PI controller and simulation parameters

The window with controller parameters shows also the parameters of the transfer function of the controlled process. The designed controller is tested by simulation of the closed-loop step response. The step changes of the setpoint and of the disturbance are supposed and corresponding parameters have to be also set. Some of these parameters have pre-set values. Pressing the button `Step response` leads to displaying the closed-loop step response. Pressing the button `New` gives the possibility to change parameter settings.

The simulated control response generated by the software is shown in Figure 9. The time behavior of the manipulated variable is shown in Figure 10. Both figures were generated for parameters shown in Figure 8. The control response is displayed for the realized, i.e. constrained control input. Displaying of the closed-loop control response helps to decide whether the designed controller can be used for control of the given process. The impact of control-input constraints on the quality of control can be judged by generating several step responses with various settings of control-input boundaries and by comparing quality criteria for each step response.

The quality of control is judged by integral quality criteria: *ISE* (2) and *IAE* (3), see Figure 11. The quality criteria serve for comparison of several designed controllers.

$$I_{IAE} = \int_0^\infty |e(t)dt| \qquad (2)$$

$$I_{ISE} = \int_0^\infty e^2(t)dt \qquad (3)$$

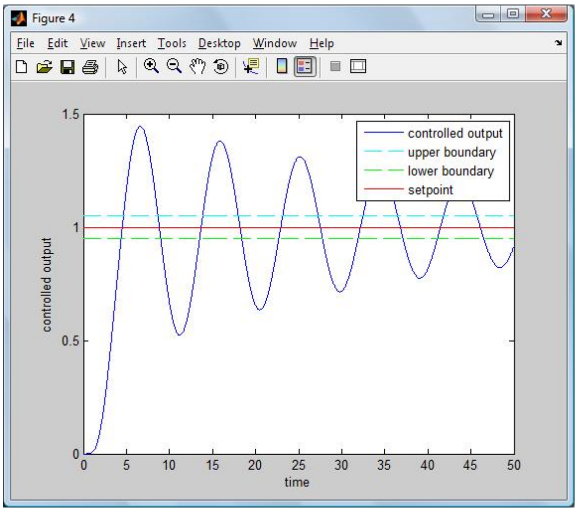

Fig. 9. Closed-loop step response generated by the software

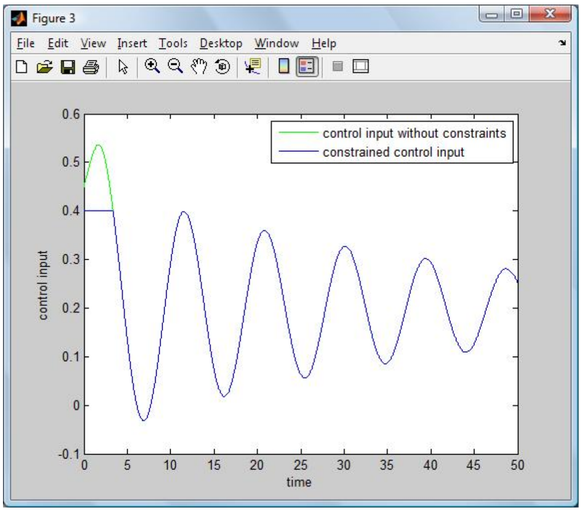

Fig. 10. Control input

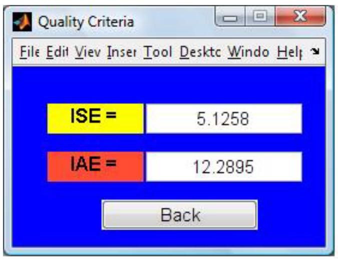

Fig. 11. Resulting quality criteria

3 CONCLUSION

PIDTOOL 1.0 was developed in the MATLAB 6.5 Simulink 5.0 Release 13 programming environment. MATLAB Graphical User Interface offered comfortable and simple tool for identification and PID controller tuning.

Using of PIDTOOL is very simple and it was tested and used up to now by students at the FCFT in two courses: Process Dynamics and Control, and Integrated Control in the Process Industry. The benefit of its using is seen especially in the course Process Dynamics and Control , which is taught for students in the study program Chemical and Environmental Engineering. The students in this course have only basic knowledge in process control. There are 13 exercises altogether in this course, each of them takes 50 minutes. The goal is to practice the process control by simulations in the MATLAB-Simulink environment. Various processes are controlled: tanks, reactors, distillation columns and heat exchangers. Using PIDTOOL enables to finish prescribed tasks in the disposable time.

PIDTOOL 1.0 is an open system. It will be further developed and the English version will be completed.

ACKNOWLEDGMENTS

The work has been supported by the Scientific Grant Agency of the Slovak Republic under grants 1/4055/07, 1/0071/09 and by the Slovak Research and Development Agency under the project APVV-0029-07. This support is very gratefully acknowledged.

4 REFERENCES

Åström, K. and T. Hägglund (1995). *PID controllers: theory, design and tuning*. 2nd Ed. Instrument Society of America, Triangle Park.

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

Čermanová, L. (2007). *Design of GUI for controller tuning* (in Slovak). Diploma thesis. FCFT STU in Bratislava.

Mikleš, J. and M. Fikar (2007). *Process Modelling, Identification, and Control*. Springer Verlag, Berlin Heidelberg.

Ogunnaike, B. A. and W. H. Ray (1994). *Process dynamics, modeling and control*. Oxford University Press, New York.

Oravec, J. (2008). *Development of software for controller tuning* (in Slovak). Bachelor project. FCFT STU in Bratislava.

Oravec, J. (2009). *Development of software for controller tuning* (in Slovak). Terminal project. FCFT STU in Bratislava.

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