Annals of DAAAM for 2010 & Proceedings of the 21st International DAAAM Symposium, Volume 21, No. 1, ISSN 1726-9679
ISBN 978-3-901509-73-5, Editor B. Katalinic, Published by DAAAM International, Vienna, Austria, EU, 2010
Make Harmony Between Technology and Nature, and Your Mind will Fly Free as a Bird
Annals of DAAAM International

PROGRAM SYSTEM FOR AUTO-TUNING IN MATLAB ENVIRONMENT

KORBEL, J[iri]

Abstract: *This contribution deals with the development of program system in Matlab environment for relay auto-tuning principles. It is developed for simple application of the auto-tuning to a controlled system of arbitrary order. This system is identified through a biased relay experiment as a first or second order transfer function with time delay. The consecutive control design is performed through a general solution of Diophantine equations in ring of proper and stable rational functions. The program system enables the control design for systems with neglected time delay or with time delay approximated by Pade. Controller parameters are tuned through a pole-placement problem as a desired multiple root of the characteristic closed loop equation.*

Key words: *Auto-tuning, biased relay, Diophantine equations, feedback loop, time delay*

1. INTRODUCTION

The industrial applications still use the PID type controllers. Unfortunatelly, the proper PID control design requires the prior knowledge of the controlled system parameters. The auto-tuning principle removes this disadvantage because it consists of two steps. The unknown controlled system parameters are identified in the first phase by the biased relay experiment. Then, the controller parameters are derived from general solution of Diophantine equation. The principle was introduced by Åström in 1984 when a symmetrical relay was used to obtain critical parameters of the controlled system.

2. BIASED RELAY IN FEEDBACK LOOP

The first phase in the auto-tuning is the identification of process parameters. It is done through a biased relay which is connected in the feedback loop instead of the controller. Typical output from the experiment can be seen in Fig.1. The process parameters are identified in the form of transfer function with time delay:

$$G(s) = \frac{K}{\left(T_i s + 1\right)^i} \cdot e^{-\Theta_i s} \tag{1}$$

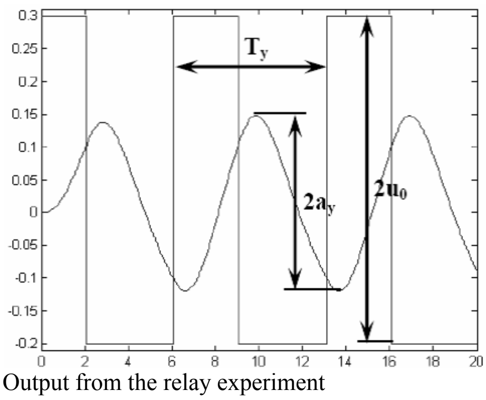

Fig. 1. Output from the relay experiment

The process gain can be computed from (Vyhlídal, 2000):

$$K = \frac{\int_0^{T_y} y(t)dt}{\int_0^{T_y} u(t)dt} \tag{2}$$

Time constant and time delay term are solved from the equations (Vítečková et al. 2004):

$$T_i = \frac{T_y}{2\pi} \cdot \sqrt{\sqrt[i]{\frac{16 \cdot K^2 \cdot u_0^2}{\pi^2 \cdot a_y^2}} - 1} \tag{3}$$

$$\Theta_i = \frac{T_y}{2\pi} \cdot \left[\pi - i \cdot \arctan \frac{2\pi \cdot T_i}{T_y} - \arctan \frac{\varepsilon}{\sqrt{a_y^2 - \varepsilon^2}} \right] \tag{4}$$

where ε is relay hysteresis.

3. CONTROLLER DESIGN

The control design is based on the fractional approach (Vidyasagar 1987), (Kučera 1993), (Prokop et al. 1997), (Prokop et al. 2002). The conversion between polynomial and R_{PS} form can be expressed as:

$$G(s) = \frac{b(s)}{a(s)} = \frac{\dfrac{b(s)}{(s+m)^n}}{\dfrac{a(s)}{(s+m)^n}} = \frac{B(s)}{A(s)} \tag{5}$$
$$n = \max(\deg(a), \deg(b)), \quad m > 0$$

The time delay term can be approximated by the Pade approximation (6) before controller synthesis.

$$e^{-\Theta s} \approx \frac{1 - \dfrac{\Theta s}{2}}{1 + \dfrac{\Theta s}{2}} \tag{6}$$

All stabilizing controllers are given by the general solution of the Diophantine equation:

$$AP + BQ = 1 \tag{7}$$

which can be expressed by:

$$P = P_0 + BZ \qquad Q = Q_0 - AZ \tag{8}$$

where P_0 and Q_0 are particular solutions of (7) and Z is an arbitrary element of R_{PS}.

4. PROGRAM SYSTEM IN MATLAB

For simple application of auto-tuning principle a program system was developed in Matlab-Simulink environment. This program enables an identification of the controlled system of arbitrary order as the first or second order transfer function with time delay. The user can choose if time delay should be neglected or approximated by Pade before control design. The program is developed with help of the Polynomial Toolbox.

Main menu of the program system can be seen in Fig. 2. The field “1” contains edit boxes where users can enter the controlled system transfer function of arbitrary order. Further, the order of identified model can be chosen in “2”. It is also necessary to select how the time delay should be treated. The area “3” contains two options. It is possible to neglect the time delay in control design, or approximate it by Pade. When all options are set, the relay experiment must be performed by pressing appropriate button in area “4”. If the experiment is successful, the program displays parameters of the identified model in “5”. Comparison of step responses of controlled and identified systems can be displayed by pressing another button in “4”. Parametrs such as duration of the relay experiment and simulation, or value of the tuning parameter, can be altered in “6”. Controller paramteres are computed by pressing button *Controller design* in “7” and are displayed in “9”. In the area “7” is also *Simulation* button which starts the Simulink environment for verification of the designed controller. If the user has selected ‘Approximate by Pade’ in “3” then the area “8” will display transfer function of the identified system after time delay approximation.

5. SIMULATION

A fourth order controlled system governed by the transfer function:

$$G(s) = \frac{1.5}{(1.5s+1)^4} \cdot e^{-3s} \tag{9}$$

was approximated as a first order model:

$$\widetilde{G}(s) = \frac{1.5}{3.9s+1} \cdot e^{-5.7s} \tag{10}$$

The control responses are shown in figures 3 and 4. The tuning parameter m has the same value in both cases. Figure 3 shows the control response when the time delay was neglected in controller synthesis. The only difference in Fig. 4 is the approximation of the time delay before the control design.

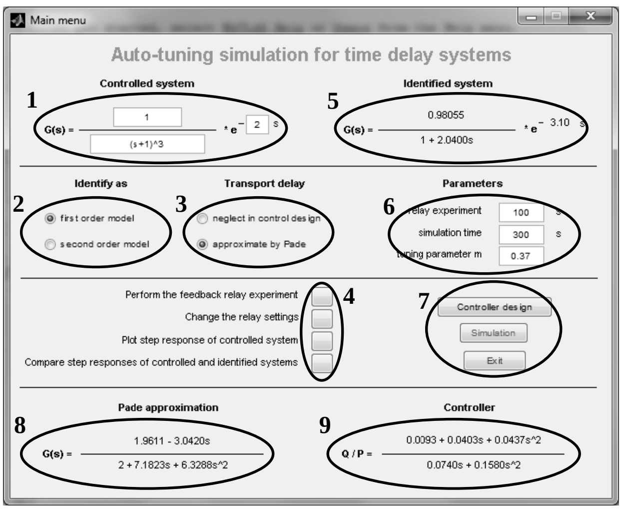

Fig. 2. Main menu of the program system

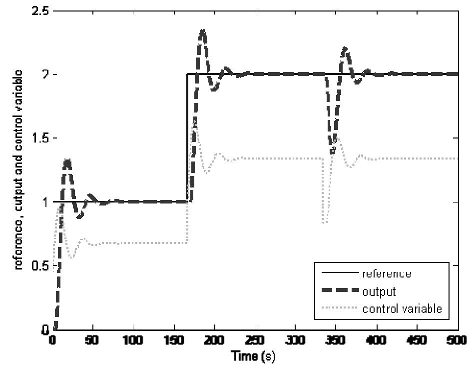

Fig. 3. Neglected time delay, m=0.19

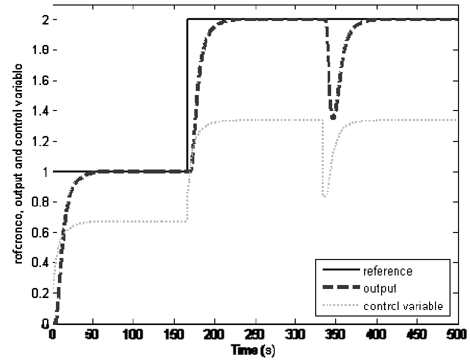

Fig. 4. Approximation of time delay by Pade, m=0.19

6. CONCLUSION

The auto-tuning methodology studied in this contribution was supported by developing the program system in Matlab environment. This program enables quick and simple design, simulation and verification of the auto-tuning approach. If the control output is not satisfactory, the user can alter the parameters of the design process and repeat the simulation very quickly. The given example showed the simplicity and efficiency of the proposed methodology.

7. ACKNOWLEDGMENT

This work was supported by the grant of Ministry of Education, Youth and Sports of the Czech Republic, MSM 708 835 2102.

8. REFERENCES

Åström, K.J. & Hägglund, T. (1984). Automatic tuning of simple regulators with specification on phase and amplitude margins. *Automatica,* vol. 20, pp. 645-651

Kučera, V. (1993). Diophantine equations in control - A survey. *Automatica*, Vol. 29, pp. 1361-75

Prokop, R. & Corriou, J.P. (1997). Design and analysis of simple robust controllers. *Int. J. Control*, Vol. 66, pp. 905-921

Prokop, R.; Husták, P. & Prokopová, Z. (2002). Simple robust controllers: Design, tuning and analysis. In *Preprints of 15th IFAC World Congress*

Vítečková, M. & Víteček, A. (2004). Experimental identification by relay methods. In *Automatizácia a informatizácia*

Vidyasagar, M. (1987). *Control system synthesis: a factorization approach.* MIT Press, Cambridge, M.A

Vyhlídal, T. (2000). Anisochronic first order model and its application to internal model control. In *ASR ‘2000 Seminar*