ROBUST CONTROLLER DESIGN BY GENETIC ALGORITHM

M. Hypiusova, S. Kajan

Institute of Control and Industrial Informatics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava, Slovak Republic

Abstract

The paper deals with the design of robust controllers for uncertain SISO systems using genetic algorithm. The genetic algorithm represents an optimisation procedure, where the costs function to be minimized comprises the closed-loop simulation of the control process and a selected performance index evaluation. Using this approach the parameters of the PID controller were optimised in order to become the required behaviour of the control process. The practical application is illustrated by the robust controller design for a Modular Servo System.

1 Introduction

For many real processes a controller design has to cope with the effect of uncertainties, which very often cause a poor performance or even instability of closed-loop systems. The reason for that is a perpetual time change of parameters (due to aging, influence of environment, working point changes *etc.*), as well as unmodelled dynamics. The former uncertainty type is denoted as the parametric uncertainty and the latter one the dynamic uncertainty. A controller ensuring closed-loop stability under both of these uncertainty types is called a robust controller. A lot of robust controller design methods are known from the literature [1], [2] in the time- as well as in the frequency domains.

The focus of this paper is to show robust PID controller design to control angular velocity of a Modular Servo System for three working points. The method is based on the Kharitonov systems considering uncertain system model with parametric uncertainties. The identified models are used for simulation of performance index in genetic algorithm. Control performance indices corresponding to robust controllers designed for three required closed-loop stability degree are compared in three working points.

2 Modular Servo System

The Modular Servo System (MSS) consists of the Inteco digital servomechanism and open-architecture software environment for real-time control experiments [11]. The measurement system is based on the RTDAC4/USB acquisition board equipped with a D/A and A/D converters. I/O board communicates with the power interface unit. The whole logic necessary to activate and read the encoder signals and to generate the appropriate sequence of the PWM pulses to control the DC motor is configured in the Xilinx® chip of the RT-DAC/USB board. All functions of the board are accessed from the Modular Servo Toolbox, which operates directly in the MATLAB Simulink environment [10].

The MSS consists of the following modules arranged in the chain: the DC motor with the generator, inertia load, encoder, magnetic brake and the gearbox with the output disk depicted in Fig. 1. The system has no got an inner feedback for dead zone compensation. The accuracy of the measured velocity is 5% while the accuracy of the angle is 0.1%. The armature voltage of the DC motor is controlled by a PWM signal v(t) excited by a dimensionless control signal in the form $u(t) = v(t)/v_{max}$.

In our experiment backlash module was not applied. The servomechanism is connected to a computer where a control algorithm is realised based on measurements of the angle and angular velocity. In our paper only the angular velocity was controlled.

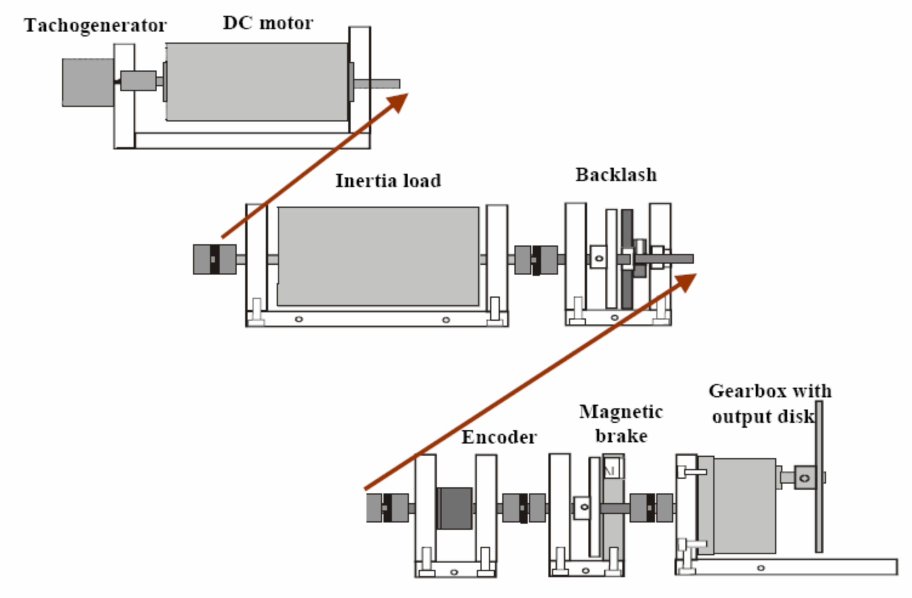

Figure 1: MSS mechanical construction

3 Robust controller design using the Kharitonov system and genetic algorithm

3.1 The Kharitonov systems

Consider three working points of the controlled system. Comparing coefficients of equal powers of s of the three transfer functions we obtain intervals of coefficients:

$$b_i \in \langle \underline{b}_i, \overline{b}_i \rangle \quad \text{and} \quad a_j \in \langle \underline{a}_j, \overline{a}_j \rangle \quad \text{for } i = 0,1,m;\ j = 1,2,\ldots,n \tag{1}$$

where $a_n \neq 0; b_m \neq 0; m \leq n$.

Interval polynomials of the system numerator and denominator respectively are given

$$\mathbf{N}(s) = \{B(s) : b_0 + \ldots + b_m s^m, b_i \in \langle \underline{b}_i, \overline{b}_i \rangle, i = 0,1,\ldots,m\} \tag{2}$$

$$\mathbf{D}(s) = \{A(s) : a_0 + \ldots + a_n s^n, a_i \in \langle \underline{a}_i, \overline{a}_i \rangle, i = 0,1,\ldots,n\} \tag{3}$$

The interval system is defined in the form

$$\mathbf{G}(s) = \left\{ \frac{B(s)}{A(s)} : (B(s), A(s)) \in (\mathbf{N}(s), \mathbf{D}(s)) \right\} \tag{4}$$

Consider $I(s)$ to be the set of closed-loop characteristic polynomials of degree *n* for the interval systems (4)

$$p(s) = A(s) + B(s) = p_0 + p_1 s + p_2 s^2 + \ldots + p_n s^n \tag{5}$$

where $p_0 \in \langle \underline{p}_0, \overline{p}_0 \rangle$, $p_1 \in \langle \underline{p}_1, \overline{p}_1 \rangle$, ..., $p_n \in \langle \underline{p}_n, \overline{p}_n \rangle$.

Such a set of polynomials is called *interval family* and we refer to $I(s)$ as to interval polynomial. The necessary and sufficient condition for the stability of the entire family is formulated in the Kharitonov's Theorem.

Theorem 1 (Kharitonov's Theorem)

Every polynomial in the family $I(s)$ is stable if and only if the following four extreme polynomials are stable:

$$K^1(s) = \underline{p}_0 + \underline{p}_1 s + \overline{p}_2 s^2 + \overline{p}_3 s^3 + \underline{p}_4 s^4 + \ldots = p^{--} \tag{6}$$

$$K^2(s) = \underline{p}_0 + \overline{p}_1 s + \overline{p}_2 s^2 + \underline{p}_3 s^3 + \underline{p}_4 s^4 + \ldots = p^{-+}$$

$$K^3(s) = \overline{p}_0 + \underline{p}_1 s + \underline{p}_2 s^2 + \overline{p}_3 s^3 + \overline{p}_4 s^4 + \ldots = p^{+-}$$

$$K^4(s) = \overline{p}_0 + \overline{p}_1 s + \underline{p}_2 s^2 + \underline{p}_3 s^3 + \overline{p}_4 s^4 + \ldots = p^{++}$$

If the coefficients p_i vary dependently, the Kharitonov's Theorem is conservative. In such a case we can use the set of Kharitonov systems as follows:

$$\mathbf{G}_K(s) = \left\{ \frac{K_B^i(s)}{K_A^j(s)} : i, j = 1,2,3,4 \right\} \tag{7}$$

where $K_B^i(s)$, i = 1,2,3,4 and $K_A^j(s)$, j = 1,2,3,4 denote the Kharitonov polynomials associated with $\mathbf{N}(s)$ and $\mathbf{D}(s)$ respectively.

Theorem 2

The closed loop system containing the interval plant **G**(s) is robustly stable if and only if each of the Kharitonov systems in $\mathbf{G}_K(s)$ is stable.

3.2 Genetic algorithm

As mentioned above, the aim of the control design is to provide required static and dynamic behaviour of the controlled process. Usually, this behaviour is represented in terms of the well-known concepts referred in the literature: maximum overshoot, settling time, decay rate, steady state error or various integral performance indices [3, 8].

Without loss of generality let us consider a feedback control loop (closed-loop) (Fig.2), where angular velocity y is the regulated variable, input voltage u is the manipulated variable, w is the reference variable of angular velocity and e is the control error ($e=w-y$).

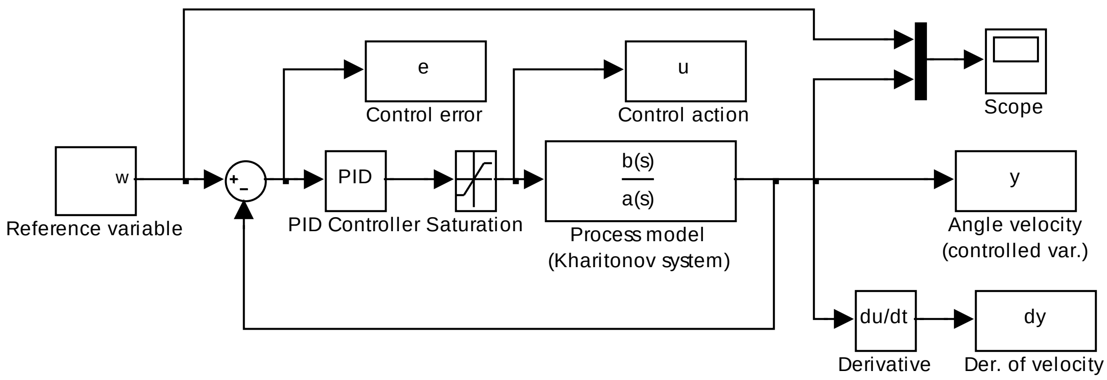

Figure 2: The control loop simulation scheme

The controller design principle is actually an optimization task - search for such controller parameters from the defined parameter space, which minimize the performance index. The cost function (fitness) is a mapping $R^n \to R$, where n is the number of designed controller parameters. The cost function can to represent sum of absolute control errors (SAE) or sum of time weighting absolute control errors (STAE) in following forms:

$$J = \sum_{i=1}^{N} |e_i| \quad = \quad \sum_{i=1}^{N} |w_i - y_i|, \tag{8}$$

$$J = \sum_{i=1}^{N} t_i \cdot |e_i| \quad = \quad \sum_{i=1}^{N} t_i \cdot |w_i - y_i|, \tag{9}$$

where t is time, w is reference variable, y is controlled output, e is control error and N is number of patterns. Fitness is represented by the cost function or in the case of control, by the modified cost function, which can be penalized for example by derivation of process output y or by overshoot of process output or by saturation of control action u.

The evaluation of the cost function consists of two steps. The first step is the computer simulation of the closed-loop time-response, and the second one is the performance index evaluation.

Genetic algorithms are described in e.g. [3, 4, 5, 6, 7, 8] and others. Each chromosome represents a potential solution, which is a linear string of numbers, whose items (genes) represent in our case the designed controller parameters. Because the controller parameters are real-number variables and in case of complex problems the number of the searched parameters can be large, real-coded chromosomes have been used.

The searched PI controller parameters are $K \in R^+$, $Ti \in R^+$. The chromosome representation in this case can be in form ***ch***$=\{K, Ti\}$.

A general scheme of a GA can be described by following steps (Figure 3):

1. Initialisation of the population of chromosomes (set of randomly generated chromosomes).

2. Evaluation of the cost function (fitness) for all chromosomes.

3. Selection of parent chromosomes.

4. Crossover and mutation of the parents → children.

5. Completion of the new population from the new children and selected members of the old population. Jump to the step 2.

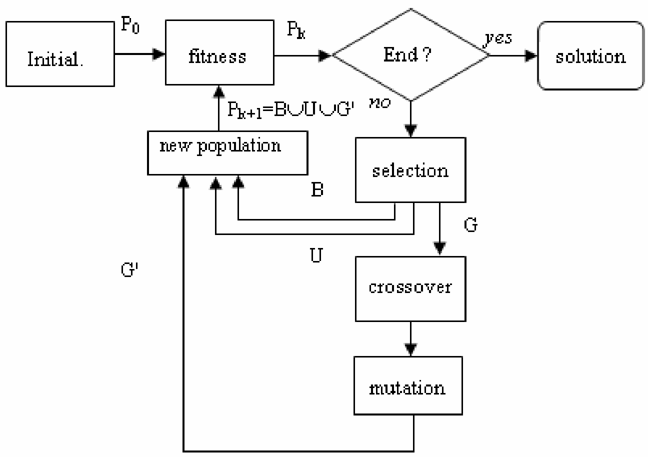

Figure 3: Block scheme of the used genetic algorithm

A block scheme of a GA-based design is in Figure 4. Before each cost function evaluation, the corresponding chromosome (genotype) is decoded into controller parameters of the simulation model (phenotype) and after the simulation the performance index is evaluated.

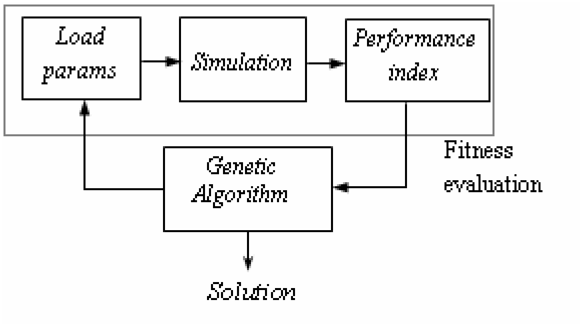

Figure 4: Block scheme of the GA-based controller design

4 Design of robust controller for Modular Servo System

4.1 System identification using the Kharitonov systems

Consider the transfer functions of an angular velocity of the Modular Servo System obtained by identification in three working points:

WP1: manipulated variable $u = 0.2$ [V]; regulated variable $y = 15$ [rad/s];

$$G_{p_1}(s) = \frac{245.8}{s + 1.089}$$

WP2: manipulated variable $u = 0.5$ [V]; regulated variable $y = 85$ [rad/s];

$$G_{p_2}(s) = \frac{248.6}{s + 1.049}$$

WP3: manipulated variable $u = 0.7$ [V]; regulated variable $y = 130$ [rad/s];

$$G_{p_3}(s) = \frac{244}{s + 1.036}$$

The Kharitonov approach uses the interval model:

$$G_p(s) = \frac{B(s)}{A(s)} = \frac{b_0}{a_1 s + 1} \tag{10}$$

where $b_0 \in \langle 225.7, 237 \rangle$, $a_1 \in \langle 0.918, 0.965 \rangle$.

The set of Kharitonov systems is reduced to only four Kharitonov systems:

$$G_{K1}(s) = \frac{225.7}{0.9183s + 1} \qquad G_{K2}(s) = \frac{237}{0.9183s + 1} \tag{11}$$

$$G_{K3}(s) = \frac{225.7}{0.9653s + 1} \qquad G_{K4}(s) = \frac{237}{0.9653s + 1}$$

Figure 5 shows the step responses of the Kharitonov systems.

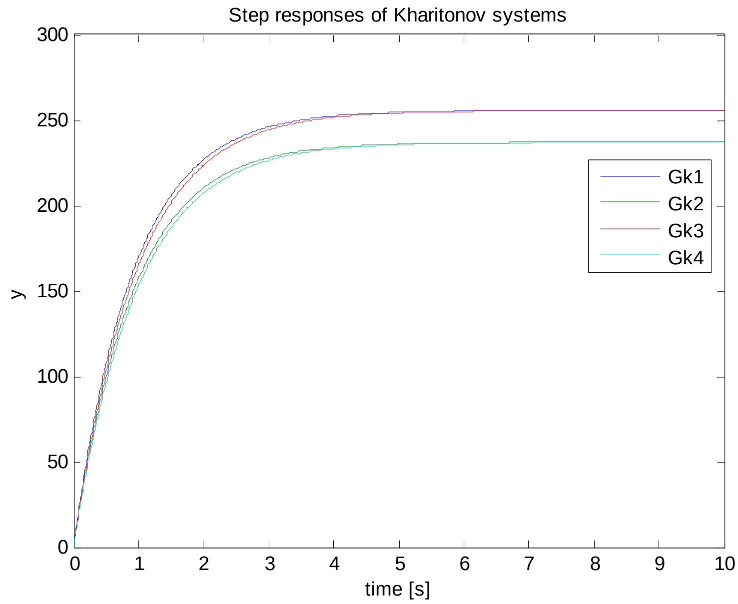

Figure 5: Step responses of the Kharitonov systems

4.2 Design of robust PI controller using genetic algorithm

We have designed a robust PI controller for the all four Kharitonov systems in the form

$$G_R(s) = K\left(1 + \frac{1}{T_i s}\right) \tag{12}$$

where the gain $K = 0.0262$ and the integration time constant $T_i = 1.5567[s]$. For optimization of PI controller parameters we used of genetic algorithm depictured in Figure 3. GA contained 30 chromosomes, tournament selection, one point crossover and additive mutation with mutation rate 0.2. For searching optimal PI controller parameters the used cost function (fitness) was considered by equation (8), which is penalize by overshoot of process output and saturation of control action. The fitness is evaluated in all four Kharitonov systems. In Figure 6 the cost function convergence during GA-runs (cost function vs. generation number) is depicted.

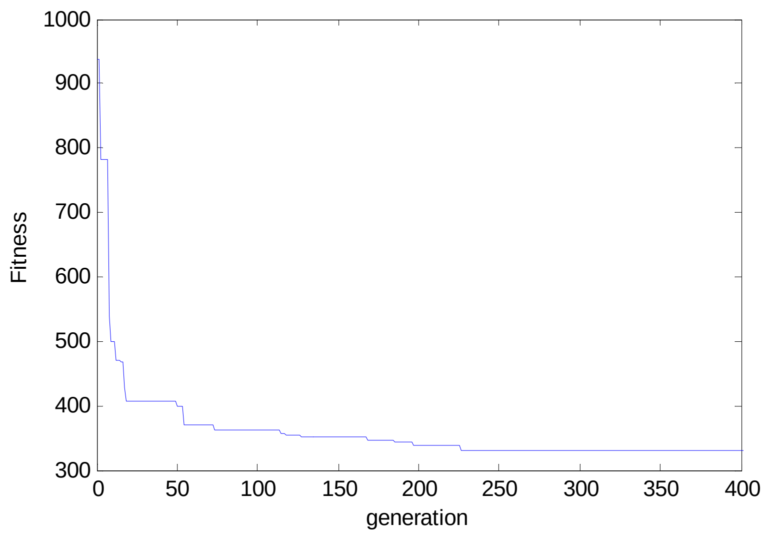

Figure 6: Evolution of the fitness function

4.3 Verification of robust PI controller in Modular Servo System

For verification of robust controller design by genetic algorithm was used a program ROBUGEN, which was created for design and verification robust controller in Modular Servo System [9]. The obtained results after these design steps of robust PI controller, which were are demonstrated in Figure 7. It is the time-response of the controlled variable (blue) after the reference signal steps (red). The control value time-response is in Figure 8. Table 1 presents comparison of criterion control quality values in three working points.

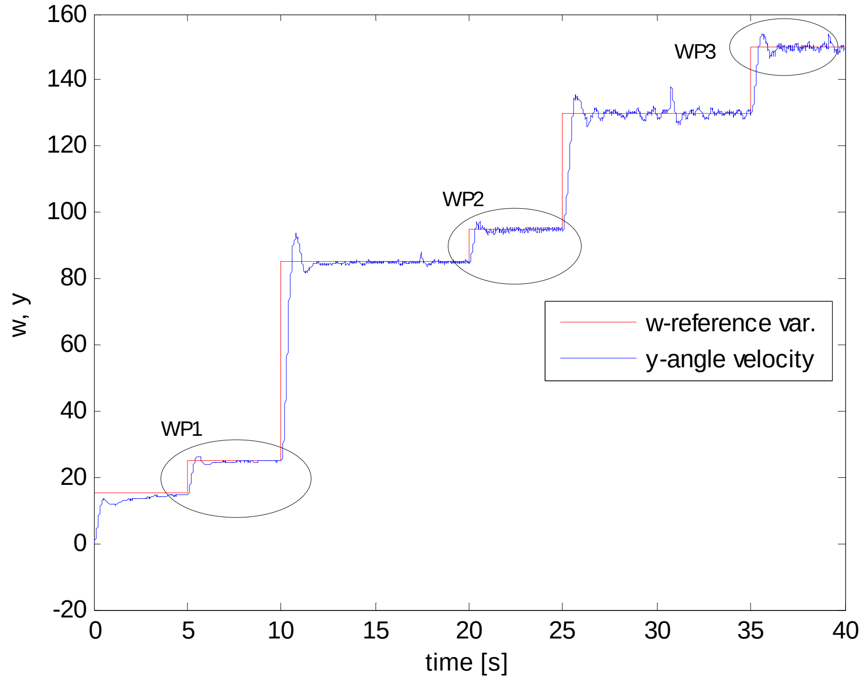

Figure 7: Time-response of the controlled system

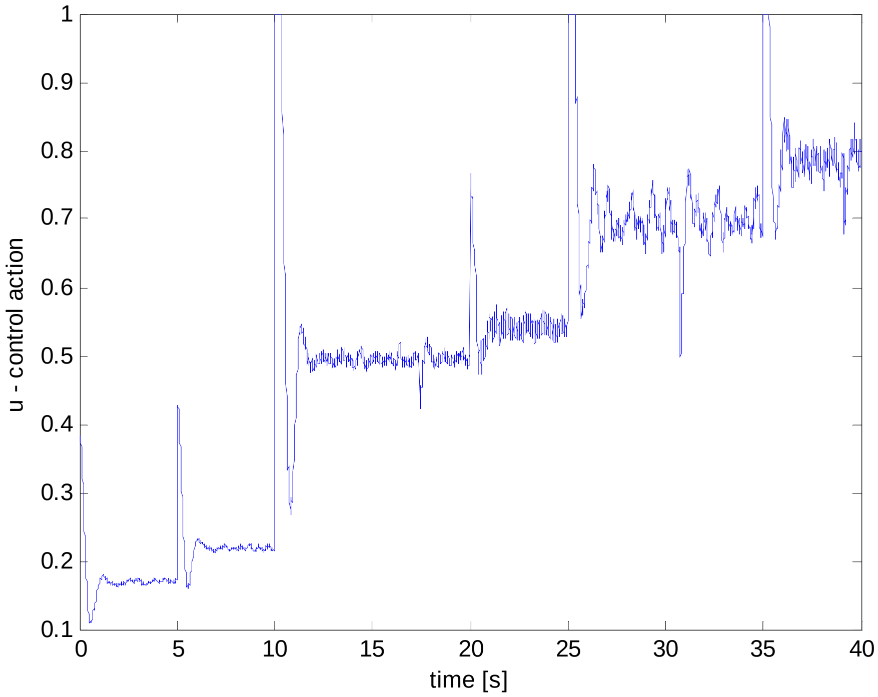

Figure 8: Control value of PI controller

TABLE 1: CRITERION CONTROL QUALITY VALUES

	WP1	WP2	WP3
SAE	418.71	436.47	874.83
Overshoot	1.4605	2.157	4.1665
Control time [s]	1.49	3.9	3.78

5 Conclusion

The main aim of this paper has been to design a robust controllers using genetic algorithm for MSS (Modular Servo System). The presented method is based on the robust stability analysis results of Kharitonov systems and guarantees robust stability of interval system which was defined by three working points. Using genetic algorithm the parameters of the PID controller were optimised in order to become the required behaviour of the control process. The results obtained by verification on MSS show the effectiveness of the proposed method.

References

[1] J. Ackerman. *Robust Control - Systems with Uncertain Physical Parameters.* Springer-Verlag London Limited, 1997.

[2] S.P. Bhattacharyya, H. Chapellat, L.H.Keel. *Robust Control: The parametric Approach*. Prentice Hall, 1995.

[3] R.C. Dorf. *Modern Control Systems*. Addison-Wesley publishing Company, 5th edition, 1990

[4] D.E.Goldberg. *Genetic Algorithms in Search, Optimisation and Machine Learning*. Addisson-Wesley, 1989

[5] K.F. Man, K.S. Tang, S. Kwong. *Genetic Algorithms, Concepts and Design*. Springer, 2001

[6] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolutionary Programs*. Springer, 1996

[7] I.Sekaj. *Evolučné výpočty a ich využitie v praxi*, Iris 2005, Bratilava (in slovak)

[8] I.Sekaj. *Genetic Algorithm Based Controller Design*, In: 2nd IFAC conference Control System Design'03, Bratislava

[9] K. Marton. *Návrh regulátorov laboratórneho modelu servosystému*, diplomová práca, FEI STU Bratislava, 2009

[10] The Mathworks. *Matlab ver. 7.1 (R14), user documentation*, 2006

[11] INTECO. *Modular Servo System, user manual*, 2007

Acknowledgments

The work has been supported by the grants agency VEGA no. 1/0544/09 and no. 1/0690/09.

Ing. Mária Hypiusová, PhD.:

Institute of Control and Industrial Informatics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava, Slovak Republic, Ilkovičova 3, 812 19 Bratislava, E-mail: maria.hypiusova@stuba.sk

Ing. Slavomír Kajan, PhD.:
Institute of Control and Industrial Informatics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology in Bratislava, Slovak Republic, Ilkovičova 3, 812 19 Bratislava, E-mail: slavomir.kajan@stuba.sk