ACTA TECHNICA CORVINIENSIS – Bulletin of Engineering
Tome VI (Year 2013) – FASCICULE 1 [January–March]
ISSN 2067–3809

[1.] Tatiana RADIČOVÁ, [2.] Milan ŽALMAN

LMPM MASTER SLAVE POSITION CONTROL WITH LUENBERGER OBSERVER USING GENETIC ALGORITHMS

[1.] SLOVAK UNIVERSITY OF TECHNOLOGY, FACULTY OF ELECTRICAL ENGINEERING & INFORMATION TECHNOLOGY, TRENČÍN, SLOVAKIA
[2.] SLOVAK UNIVERSITY OF TECHNOLOGY, FACULTY OF ELECTRICAL ENGINEERING & INFORMATION TECHNOLOGY, BRATISLAVA, SLOVAKIA

ABSTRACT: Linear motors tend to be indispensable at present. Whether they are utilize in health service or in automation industry. It is certain that companies is always looking for something 'more' and linear motors have it. Therefore the aim of this paper is to find better solution for introduced task. How to achieve higher precision in LMPM position control? How to adjust optimal controller parameters? This paper contains answers for more than these questions and in addition compares in more detail Pole-placement method with genetic algorithm, as well.
KEYWORDS: Master Slave Control, Luenberger Observer, Linear Motor, Genetic Algorithm, Lead Compensator, Pole placement method

INTRODUCTION

Nowadays, linear motors types are still more and more preferred thanks to their matchless features. Although in comparison with rotary motors, they are henceforward financially demanding. It can be mentioned that a high speed trains, such as Maglev or Trans-rapid became famous by implementation these kind of motors. Linear motors however occur in various sectors, whether in electro-technical or electronic production (drives for operating and position engineering, drives into machine tools ...).

This paper will be focused on position servo-drive control design of LMPM in consideration of control performance comparing Genetic Algorithm (GA) with Pole-placement (PP) method. It has to be mentioned as well that 3D Master slave control with Luenberger observer will be applied.

Observers are algorithms that combine sensed signals with other knowledge of the control system to produce observed signals. In some cases, the observer can be used to enhance system performance. It can be more accurate than sensors or can reduce the phase lag inherent in the sensor. Observers can also provide observed disturbance signals, which can be used to improve disturbance response. In other cases, they can reduce system cost by augmenting the performance of a low-cost sensor so that the two together can provide performance equivalent to a higher cost sensor. In the extreme case, observers can eliminate a sensor altogether, reducing sensor cost and the associated wiring [3]. Consequently among various observers, Luenberger observer was chosen to enhance the accuracy and reduce the sensor-generated noise.

Idea of evolutionary computing was introduced in the 1960s by I. Rechenberg in his work "Evolution strategies". His idea was then developed by other researchers. GAs were invented by John Holland and developed by him and his students and colleagues. This lead to Holland's book "Adaption in Natural and Artificial Systems" published in 1975. In 1992 John Koza has used genetic algorithm to evolve programs to perform certain tasks. He called his method "genetic programming". During reproduction, first occurs recombination (or crossover). Genes from parents form in some way the whole new chromosome. The new created offspring can then be mutated. Mutation means, that the elements of DNA are a bit changed. These changes are mainly caused by errors in copying genes from parents [8].

GA is one of the most famous and the most used representatives of evolutionary computing techniques with wide range of application [1]. Control performance possesses highly important function in servo-drives that is why we took advantage of GA to improve the overall performance. Finally results gained from the GA are compared to these learned from the controller designed by Pole-placement method with lead-compensator described in (Radicova, Zalman) [2]. Main idea of designing controller parameters using GA has been publicly adopted in the 1990's, but remains popular in the present as well, which is proven by number of papers in relevant journals [6],[7]. Interesting is also attempt of PI position controller design of SMPM drive by Khater and others [5].

3D MASTER SLAVE SERVO-DRIVE

A. Master slave control

Master slave control can be assign to the status control or model control [4.]. Its significant advantage is the inutility of knowing the exact mathematical model of controlled system S. The quality model of regulating system is highly *sufficient providing that you are familiar with the scale of main parameters.*

Master serves as a generator of control state variables and surprisingly the control vector can be greater than number of measured variables. Its task is to generate desired state value curves - control vectors which shape can be rectangular, trapezoidal or sinusoidal either 3-dimensional or 4-dimensional. In this paper is used 3D master slave generator and it generates state values of the position, speed and acceleration (Figure 2).

Slave contains controllers of the state values.

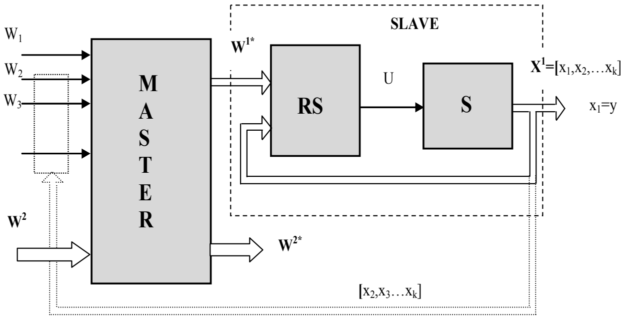

Figure 1. Master Slave control system structure

RS represents regulating system and S means controlled system.

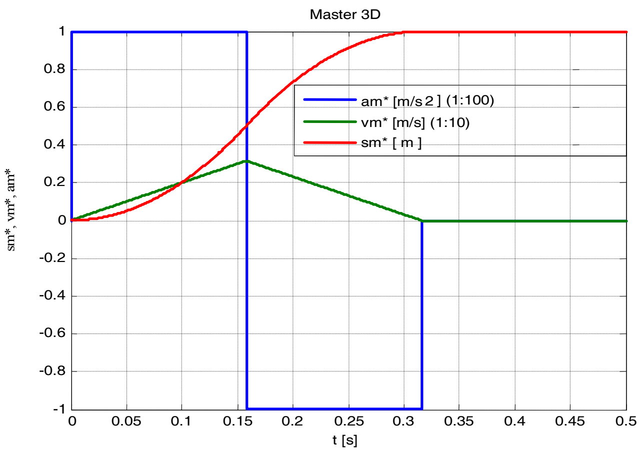

Figure 2. Time responses of Master slave output values

B. Precorrection constants

Undeniable parts in Master slave control are precorrection constants. Their task is to enhance position accuracy and consequently lower the position error.

Calculating precorrection coefficients (K_1, K_2) starts from the condition for feed forward control, however force generator GF dynamics is not considered.

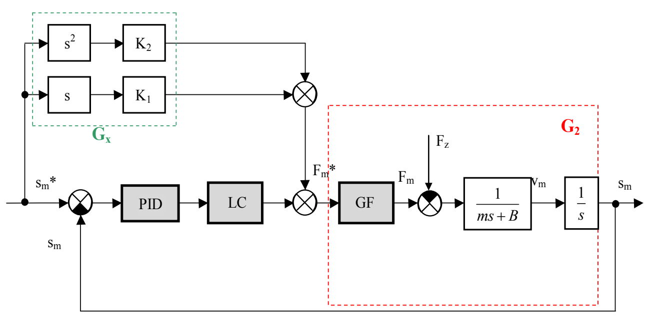

Figure 3. Force Position servo-drive block diagram with PID structure and marked precorrection Master-3D (PID-proportional-integral-derivative controller, LC-Lead Compensator, GF-Force generator,)

$$G_x(s) = \frac{1}{G_2(s)}$$

$$G_2(s) = \frac{1}{ms+B}\frac{1}{s} = \frac{1}{ms^2+Bs} \qquad (1)$$

$$G_x(s) = \frac{1}{G_2(s)} = ms^2 + Bs$$

$$K_1 = B;\ K_2 = m$$

POSITION CONTROL

A. Position servo-drive, implementation block scheme

Position servo-drive can be performed by various algorithms. PID algorithm with lead compensator is applied, referring to the article [9]. The main asset is to execute synthesis of PID algorithm with lead compensator and evaluate better parameters design method. Pole-placement method and GA will be used as was presented in introduction. The entire position servo-drive structure may be seen in the Figure 4.

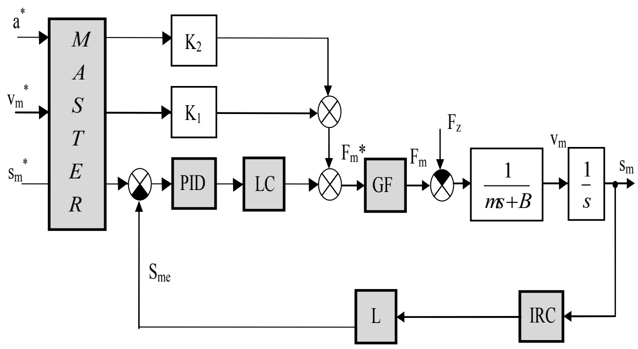

Figure 4. Entire diagram of position servo-drive (PID-proportional-integral-derivative controller, LC-Lead Compensator, GF-Force generator, L-Luenberger observer, IRC-Incremental sensor)

B. Force generator

Force generator GF (Figure 4) is the most important block of linear servo-drive control structure, which works on a principal of vector frequency-current control synchronous motor with PM control (Figure 5).

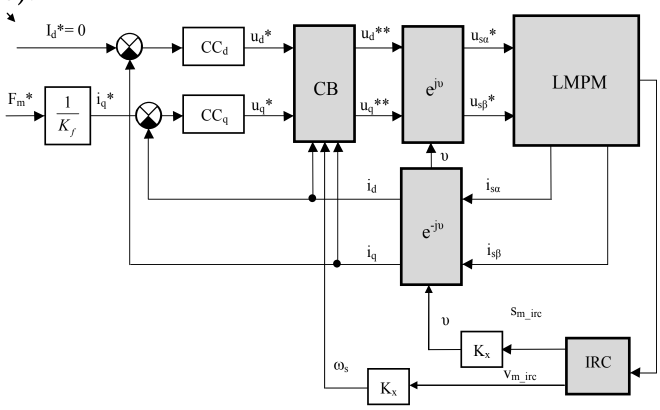

Figure 5. Force generator LMPM structure

As can be seen it contains blocks of Park's transformations, compensation block, IRC sensor, two current controllers (CC_d, CC_q) and block LMPM - particular servo-drive realized by equations

$$u_d = R_s i_d + \frac{d\psi_d}{dt} - \omega_s \psi_q$$
$$u_q = R_s i_q + \frac{d\psi_q}{dt} + \omega_s \psi_d$$
$$\psi_f = L_{md} i_f \qquad (2)$$
$$\psi_d = L_d i_d + \psi_f$$
$$\psi_q = L_q i_q$$
$$F_m - F_z = m\frac{dv_m}{dt}$$
$$\omega_s = K_x v_m \; ; \qquad \upsilon = K_x s_m$$

Equation (3) represents a relation between rotary and linear parameters issued from the physical interpretation.

$$v_m = 2\tau_p f_s \qquad (3)$$

τ_p - Pole spacing [m]
f_s - Power supply frequency [Hz]
Then generally holds the equation (4).

$$K_x = \frac{\omega_s}{v_m} = \frac{\pi}{\tau_p} \qquad (4)$$

C. Luenberger observer

Luenberger observer is the observer of velocity and acceleration. In general it may contain different algorithm structures for observing velocity and acceleration. In this paper is chosen PID algorithm for controlling the third order system however.

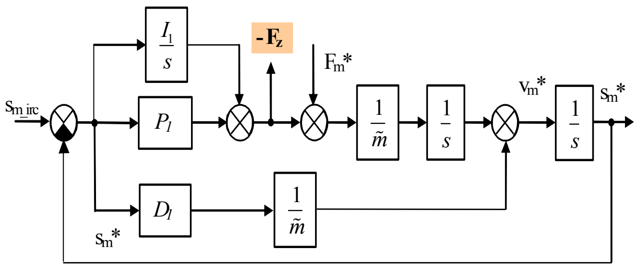

Figure 6. Luenberger observer block diagram

Pole-placement method is applied. It compares denominator of close-loop system N(s) with desired denominator $N_D(s)$ by equal power.

$$N(s) = s^3 + \left(\frac{D_1}{\tilde{m}}\right)s^2 + \frac{P_1}{\tilde{m}}s + \frac{I_1}{\tilde{m}}$$
$$N_D(s) = \left(s^2 + 2\xi_1\omega_{01}s + \omega_{01}^2\right)\left(s + k_1\omega_{01}\right) \qquad (5)$$
$$N(s) = N_D(s)$$
$$P_1 = \tilde{m}\omega_{01}^2\left(2\xi_1 k_1 + 1\right)$$
$$I_1 = \tilde{m}k_1\omega_{01}^3$$
$$D_1 = \omega_{01}\left(2\xi_1 + k_1\right)\tilde{m}$$

Parameters setup variables ξ_1, k_1 and ω_{01} are further explained in the Table III.

CONTROLLER DESIGN METHODS

The comparison of two controller design methods will be presented.
PID controller is the most used controller in praxis and it contains 3 parallel connected sub-circuits. The first sub-circuit is proportional, which multiply controller input value with adjustable coefficient.
The second parallel sub-circuit integrates and the third parallel sub-circuit derivates controller input value.
The dynamics of force generator is not considered in the synthesis of PID controller. Pole placement method is suitable for PID controller design, but it is more or less impossible to design parameters of PID with lead compensator together. According to that genetic algorithm is applied.

A. Pole-placement method

Pole placement is one of the most widely used methods of controller design. It compares denominator of close-loop system N(s) with desired denominator $N_D(s)$ by equal power. Accordingly, controller parameters are designed (P, I, D), however force generator GF dynamics is not considered.

$$N(s) = N_D(s)$$
$$N(s) = s^3 + \left(\frac{D+B}{m}\right)s^2 + \frac{P}{m}s + \frac{I}{m}$$
$$N_D(s) = \left(s^2 + 2\xi\,\omega_0 s + \omega_0^2\right)\left(s + k\omega\right) \qquad (6)$$
$$P = m\omega_0^2\left(2\xi k + 1\right)$$
$$I = mk\omega_0^3$$
$$D = \omega_0\left(2\xi + k\right)m - B$$

Parameters setup variables ξ, k and ω_0 are further explained in the Table III.
Lead compensator coefficients are design by well-known method using relation (lead - lag).

$$G_{LC} = \frac{aT_1 s + 1}{T_1 s + 1} \qquad (7)$$

The lead compensator design is not the main purpose of this paper and you can find it in (Radičová, Žalman) [2]. A task to design parameters for PID controller together with lead compensator by Pole-placement method led to analytically unsolvable problem. Therefore, another solution for this task had to be found.
However, the design of PID and lead compensator parameters were continuous, realization was discreet.

B. Genetic algorithm

GA is one of the mostly used representatives of evolutionary computing. This algorithm is based on finding optimal solution (optimal structure and controller parameters) for the given problem. Accordingly, the base rule for success is precise fitness function design. Hence the fitness function represents minimization of position error using the following

$$Fitness = \sum|e| + a\sum|dy| \qquad (8)$$

As a solving tool Genetic algorithm toolbox was used [10.]. It is not the standard part of MATLAB distribution. The Toolbox can be used for solving of real-coded search and optimization problems. Toolbox functions minimize the objective function and maximizing problems can be solved as complementary tasks as well.
The process of searching is adjusted very sophistically.

First of all, a random population is generated with a predefined number of chromosomes in one population within prescribed limits for controller parameters (for particular values see Table I).

TABLE I. USED PARAMETERS OF GENETIC ALGORITHM OPTIMIZATION

Number of generations	100
Number of chromosomes in one population	30
Number of genes in a string	3
Parameter "a" weight	0.6

Then two best strings according the fitness function were selected to the next generation. Bigger number of strings was selected to the next generation by tournament. Then number of crossovers and mutations are applied to the population to achieve bigger chances to reach the global optimum.

This algorithm, using methods mentioned above, is able to design 6 parameters for PID controller (P, I, D) and lead compensator (a_1, a_2, b_2), as well. Eq. 9 represents transfer function for lead compensator and PID controller.

$$G_{LC}(z) = \frac{a_1 z + a_2}{z + b_2}$$
$$G_{PID}(z) = P + I\frac{Tz}{z-1} + D\frac{z-1}{Tz} \quad (9)$$

Figure 7 shows algorithm progress, how was fitness function searching for optimal results.

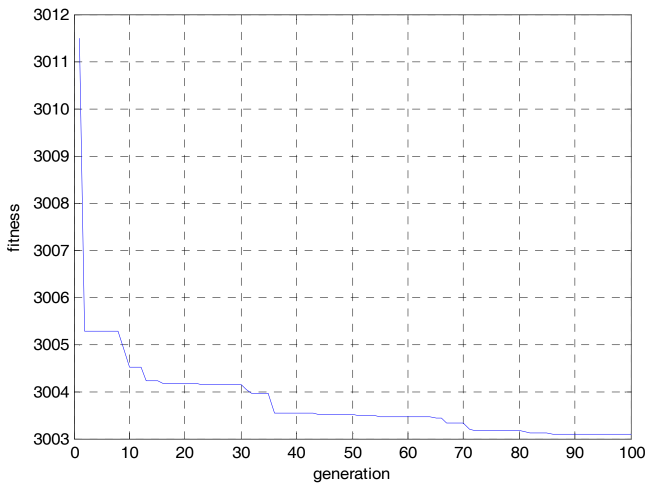

Figure 7. Convergence of genetic algorithm (GA)

RESULTS

Genetic algorithm is a powerful tool for parameters controller optimization purposes because it is able to find the solution where the other methods fail.

The main purpose was to find method how to design controller parameters as simply as possible. Genetic algorithm GA offered co-equal results as Pole-placement PP in 3D Master slave rectangular control (Figure 11). However, the major positive effect is that GA was able to design 6 parameters in discrete form at once.

In edition it has to be mentioned that following experiments were performed on the simulation model in Matlab Simulink environment using Luenberger observer and precorrection constants.

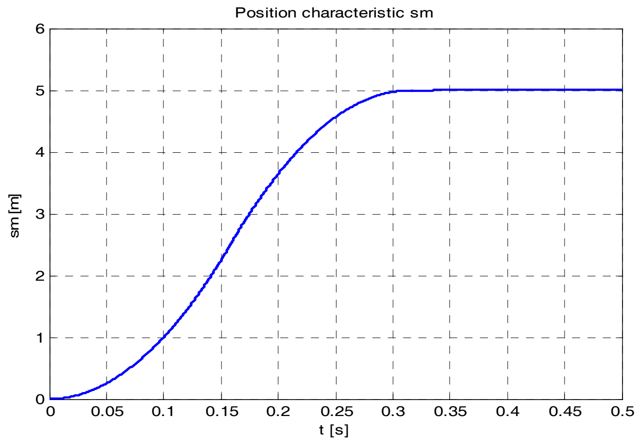

Figure 8. The time response of position

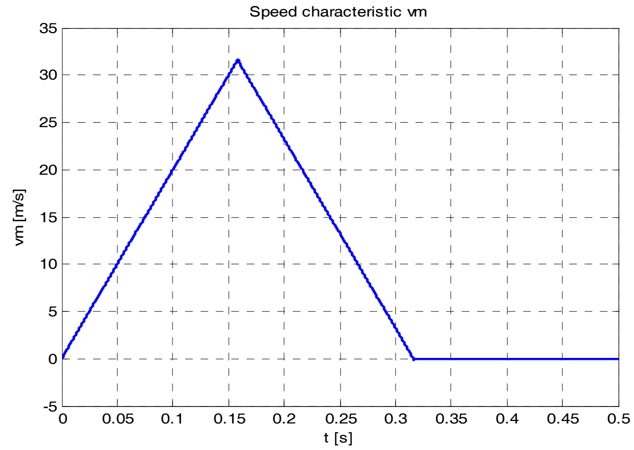

Figure 9. The time response of speed

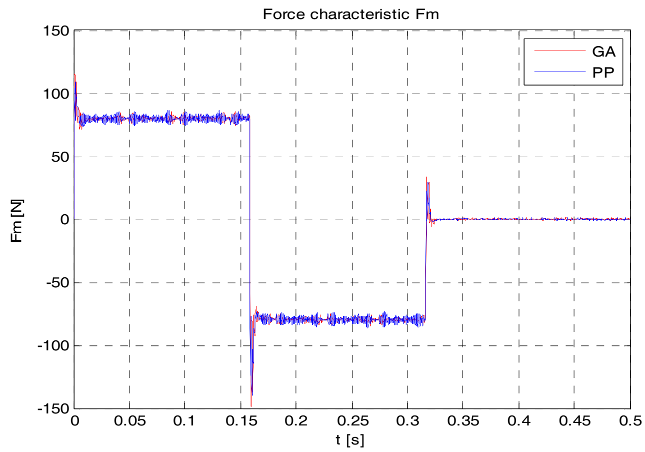

Figure 10. The time response of force

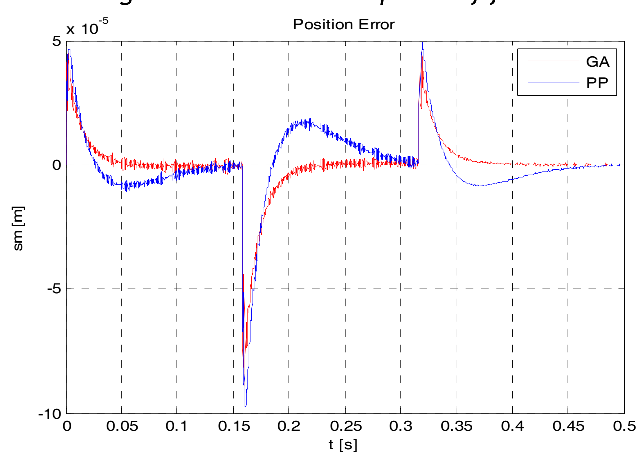

Figure 11. The time responses of position error comparing GA and PP approach

TABLE II. COMPARISON OF PARAMETERS OBTAINED BY TWO DIFFERENT METHODS

	P	I	D	a_1	a_2	b_2
GA	9487	2837	149	5.86	-0.81	0.61
Pole-placement	4737	99220	75.39	20	-0.07	-0.003

TABLE III. TABLE OF ACRONYMS

Acronym	Meaning	Value
T	Sampling period	0.2 ms
Parameters for PID controller		
ξ	Damping index	1
k	Shift pole index	1
ω_0	Bandwidth	$2\pi f_0$
f_0	Frequency	10 Hz
Parameters for Luenberger observer		
ξ_1	Damping index	1
k_1	Shift pole index	10
ω_{01}	Bandwidth	$2\pi f_0$
f_{01}	Frequency	10 Hz
Parameters for precorrection		
K_1	Precorrection constant	B = 0.01 kg.s-1
K_2	Precorrection constant	m = 0.4 kg
Parameters for IRC sensor		
N	Resolution	2 μm

CONCLUSIONS

Genetic algorithm toolbox in connection with MATLAB is a very powerful tool for optimization and search problems. The numerical results and provided figures show that within 100 generations, a solution better than analytical (Pole placement method) was found. At this point it can be said that genetic algorithm is capable to design more than 3 parameters in comparison with Pole placement method what is a significant contribution in this area. Using positive effect of precorrection and Luenberger observer led to the achievement of more precise positioning of LMPM (Figure 11). Eventually, it can be observed that PID controller with lead compensator parameters designed by genetic algorithm possess significant impact on positioning precision.

Acknowledgment

This work was supported by the Slovak Research and Development Agency under the contract No. VMSP-II-0015-09.

REFERENCES

[1.] I. Sekaj, Evolučné výpočty a ich využitie v praxi. Iris, Bratislava 2005.

[2.] T. Radičová, M. Žalman., "Master-slave position servo-drive design of aircore linear motor with permanent magnets", AT&P Journal Plus 1/2010

[3.] G. H. Ellis, Observers in Control Systems, Academic Press, 2002

[4.] M. Žalman, Akčné členy, STU Bratislava 2003

[5.] F. Khater, A. Shaltout, E. Hendawi, M. Abu El-Sebah, "PI controller based on genetic algorithm for PMSM drive system," Industrial Electronics, 2009. ISIE 2009. IEEE International Symposium pp.250-255, 5-8 July 2009 doi: 10.1109/ISIE.2009.5217925, URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5217925&isnumber=5213059

[6.] J. Solano, D. I. Jones, "Parameter determination for a genetic algorithm applied to robot control," Control, 1994. Control '94. International Conference on , vol.1, no., pp.765-770 vol.1, 21-24 March 1994 doi: 10.1049/cp:19940229, URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=327048&isnumber=7757

[7.] Y.P. Wang, D. R. Hur,H. H. Chung, N. R. Watson, J. Arrillaga, S. S. Matair, "A genetic algorithms approach to design an optimal PI controller for static VAr compensator," Power System Technology, 2000. Proceedings. PowerCon 2000. International Conference on , vol.3, no., pp.1557-1562 vol.3, 2000 doi: 10.1109/ICPST.2000.898203, URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=898203&isnumber=19429

[8.] Introduction to Genetic Algorithms, Internet article, http://www.obitko.com/tutorials/genetic-algorithms/biological-background.php

[9.] Zhou Yun-fei, Song Bao, Chen Xue-dong, "Position/force control with a lead compensator for PMLSM drive system", Springer-Verlag London Limited 2005, 18 November 2005

[10.] I.Sekaj, M.Foltin, Matlab toolbox – Genetické algoritmy. Konferencia Matlab 2003. Praha, 2003

ACTA TECHNICA CORVINIENSIS – BULLETIN of ENGINEERING

ISSN: 2067-3809 [CD-Rom, online]