ELECTRICAL ENGINEERING

ELEKTROS INŽINERIJA

Speed Control for DC Motor Drive based on Fuzzy and Genetic PI Controller – A Comparative Study

N. Ozturk

Gazi University, Faculty of Technology, Electrical&Electronics Engineering Department,
Ankara, 06500, Turkey, phone: +90 312 2028530, e-mail: ozturk@gazi.edu.tr

Introduction

A Proportional–Integral (P-I) controller is a generic control loop feedback mechanism, which is widely used in industrial control systems due to that its structure is simplex and it can almost ensure zero error in steady-state when its gains are properly tuned. A PI controller calculates an 'error' value as the difference between the desired set point and the measured output process variable. The conventional PI controller has fixed gains and is based on the mathematical model of the system being controlled. Moreover, a disadvantage of the conventional PI controller is that for optimum output quality and performance (overshoot, rise time and settling time), the controller gains must be tuned according to the nature of the system. Unfortunately, it has been quite difficult to tune the gains properly of PI controllers because many industrial plants are often burdened with problems such as high order, time delays and nonlinearities [1]. An alternative method for the speed control of a DC motor is based on fuzzy logic controller (FLC). While conventional controllers depend on the accuracy of the system model and parameters, FLCs use a different approach. Instead of using a system model, the operation of a FLC is based on heuristic knowledge and linguistic description [2]. The advantage of fuzzy control is that it can integrate the knowledge of anthropologists into the design process of controllers, without the need of the design process of controllers, without the need of accurate mathematical models [3]. However, building a FLC from a ground-up may not provide good results or sometime even a worse result than a conventional controller if there is not enough knowledge of the system. Therefore, the performance of a FLC can be improved by adjusting the rules, scaling coefficients and membership functions [4].

GA is a stochastic global adaptive search optimization technique based on the mechanisms of natural selection. In recent years, GA has been recognized as an effective and efficient technique to solve optimization problems compared with other optimization techniques [5]. A controller is decomposed into a set of parameters, which the GA attempts to optimize by using simulation based fitness evaluation of candidate controllers in the closed-loop systems [6].

This study applies genetic PI control to design a fast-response controller and improve traditional PI controllers with poorer response. The results of applying the genetic PI controller to the DC motor drive system have been compared to those obtained by the application of a fuzzy controller. The computer simulation results show that the genetic PI controller provides improved dynamic performance than the fuzzy controller. It is also observed that the genetic PI controller shows better transient performance when the system parameters are changed.

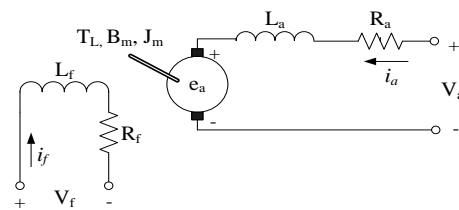### Mathematical models of a separately excited dc motor

Fig. 1. Equivalent circuit diagram of a SEDM

The equivalent circuit diagram of the separately excited DC motor (SEDM) is shown in Fig. 1. In such a DC motor, the field windings are separately excited from an external dc source [7]. Hence, the current in the field windings is independent of the current in the armature windings. The equations describing the dynamic behavior of the SEDM are given by the following equations. The voltage equations of DC motor is expressed in matrix form:

$$\begin{bmatrix} V_a \\ V_f \end{bmatrix} = \begin{bmatrix} R_a + PL_a & 0 \\ 0 & R_f + PL_f \end{bmatrix} \begin{bmatrix} i_a \\ i_f \end{bmatrix} + \begin{bmatrix} e_a \\ 0 \end{bmatrix}, \qquad (1)$$

where R_a, L_a are motor armature resistance and inductance, R_f, L_f are motor field resistance and inductance, V_a, V_f are armature and field coil voltages, i_a, i_f are armature and field coil currents, e_a is electromotive force (EMF) and P is the differential operator (d/dt)

$$e_a = k_{af} i_f \omega_m, \qquad (2)$$

where k_{af} is the armature-field mutual inductance and ω_m is the rotor electrical speed in radian.

Solving (1) yields the currents i_a and i_f, respectively. The developed torque T_e is

$$T_e = k_{af} i_f i_a. \qquad (3)$$

In the constant torque region, the speed of a SEDM could be varied from zero to rated speed mainly by varying armature voltage V_a, and hence controlling armature current i_a in (3). The ω_m can be calculated using the mechanical equation expressed in (4)

$$P\omega_m = (T_e - T_L - B_m \omega_m)/J_m, \qquad (4)$$

where B_m is the frictional coefficient and J_m is the moment of inertia of the rotor.

It is learnt from (3) that for a given constant i_f, it is only required to control armature current i_a in order to control the speed of the motor. The DC motor features under study are listed in Table 1.

Table 1. Motor features

Motor features	
Features	**Values**
Power (P)	5 HP
Maximum speed	1220 rpm
Field and armature resistance	84.91 Ω-0.4832 Ω
Field and armature inductance	13.39 H-0.006763 H
Field armature inductance	0.7096 H
Maximum current	16.2 A
Friction coefficient (B)	0.007032 N.m.s
Inertia (J)	0.2053 kg.m^2

Control techniques of SEDM

A. Proportional-Integral (PI) Controller. In control engineering, a PI Controller is a feedback controller which drives the plant to be controlled with a weighted sum of the error and the integral of that value. It is certainly the most commonly used control algorithm in industry. The main reason is its relatively simple structure, which can be easily understood and implemented in practice [6]. PI controller can be represented by

$$i^*(t) = K_P e(t) + K_i \int_0^t e(t) dt, \qquad (5)$$

where $i^*(t)$ is the reference armature current required by the motor, K_p is the proportional gain, K_t is the integral gain and $e(t)$ is the speed error signal between the desired and measured output speed. The speed error signal at t-th instant is given by

$$e(t) = \omega^* - \omega(t). \qquad (6)$$

It can be noticed that the control signal depends on the present value of the error signal as well as its previous values. In the computer simulation of a PI controller, (5) becomes

$$i^*(t) = K_P e(t) + K_i T \sum_0^t e(t), \qquad (7)$$

where $T > 0$ is the sampling period.

B. Fuzzy Logic Controller. Fuzzy control is one of the useful control techniques to deal with uncertain and ill-defined nonlinear systems. Control actions of the fuzzy controller are realized by running some linguistic rules, which also contribute to a significant interpretability of the controller [8]. The design of fuzzy controller generally includes three steps: fuzzification, inference engine and defuzzification [3]. These three steps are shown in Fig. 2. Fuzzification converts crisp input variables into fuzzy variables, inference engine library converts these linguistic values into fuzzy sets of fuzzy variables and finally defuzzification is used to convert fuzzy value into identifiable controlled variable for the controlled system
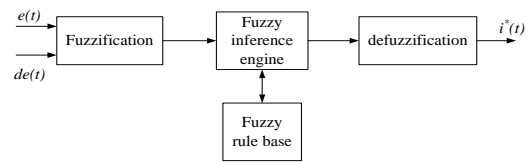

Fig. 2. Block diagram of a fuzzy controller

The process of the controlling the speed of the SEDM is a two-input and single output system. The inputs and the output of the speed controller are: the difference of speed command ω^* and actual speed ω, $i^*(t)$ is the output of the speed controller, which is the reference armature current for the hysteresis current control. Thus, the speed error $e(t)$ is taken as an input and the change of speed error $de(t)$ as the other input. So, two inputs for every sampling instant are:

$$e(t) = \omega^* - \omega(t), \qquad (8)$$

$$de(t) = (e(t) - e(t-1))/T, \qquad (9)$$

where T is the sampling period, .(t) and .$(t-1)$ are actual and previous time value, respectively.
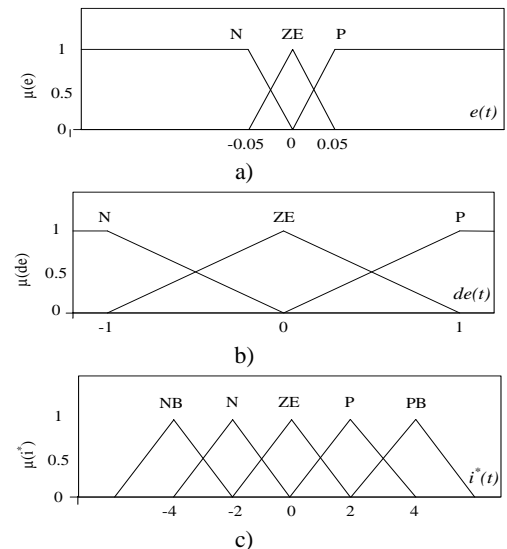

Fig. 3. Fuzzy sets on the universe of discourse for (a) speed error $e(t)$ (b) speed error change $de(t)$ (c) reference current $i^*(t)$

44

In this study, three fuzzy sets of N, ZE, P are chosen for e and de and five fuzzy sets of NB, N, ZE, P, PB are chosen for i^*, reference current command. Input and output fuzzy sets are symmetrical triangle-type membership functions. The input and output membership functions of the PI fuzzy controller are shown in Fig. 3.a, 3.b and 3.c, respectively.

The value of membership degree μ, for each fuzzy variable ranges between 0 and 1. The speed error, e, for the designed fuzzy speed controller is at the range from -0.05 to 0.05. Yet, error change variable, de, is at the range from -1 to 1 and the range of the output i^*, is set between -4 to 4.

The key point for the fuzzy controller is the establishment of the knowledge and the rule base, because they are established according to the experience of expert system [2]. Fuzzy rules are a collection of linguistic statements that describe how the fuzzy inference system (FIS) should make a decision regarding classifying an input or controlling an output [9]. A rule is expressed with the form of if-then and in this study nine rules are included in the rule base. The control rule table is given in Table 2.

Table 2. Rule base for fuzzy speed controller

i^*		de		
		N	ZE	P
	N	NB	N	ZE
e	ZE	N	ZE	P
	P	ZE	P	PB

Note: e and de are input variables. i^* is output variable.

According to the rule base given in Table 2, description of a rule in this study can be expressed in the following form: If error is positive *(P)* and error change is positive *(P)* then control signal is positive-big *(PB)*. The overall drive system used in this article is composed of a speed controller (genetic PI and fuzzy PI), a hysteresis band current controller, a current limiter and a DC-DC converter. For a PI-type fuzzy controller, because of having scaling problems in fuzzying the integral of the error signal, in the literature applications, error and error change signals are used as controller inputs and in the output, by obtaining the change of the control signal, PI feature is provided to the fuzzy controller [10]. Fuzzy PI control system is shown in Fig. 4.
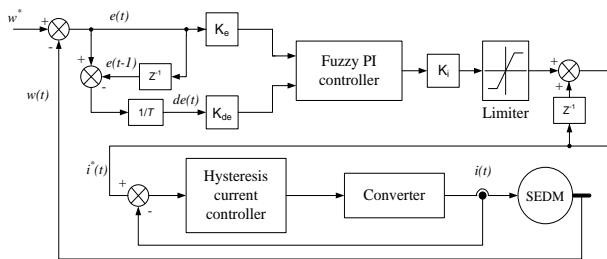

Fig. 4. Fuzzy PI control systems developed in this study

In this control strategy, in addition to fuzzy speed controller, a hysteresis current controller is inserted into the system in order to improve the performance of DC motor drive. In a hysteresis current controller (HCC), switching patterns required for DC-DC converter circuit are generated by comparing the actual current being drawn by the motor with the reference current. A positive pulse is generated if the actual current is less than the reference armature current, whereas a negative pulse is produced if the actual current exceeds the reference current [11]. Developments in the output quality and performance (overshoot, rise time, and settling time) have been achieved by applying the both control strategy. In the defuzzification process where the output of the fuzzy sets is transferred into identifiable value, the weighted average method is used in (10)

$$i^* = \frac{\sum_{i=1}^{n} u_i \mu(u_i)}{\sum_{i=1}^{n} \mu(u_i)}, \qquad (10)$$

where i^* is the crisp output of the fuzzy controller, u_i is the i-th rule weight and $\mu(u_i)$ is the membership value of the output membership function corresponding to the i-th control rule.

Genetic algorithms

The genetic algorithm is a stochastic global adaptive search optimization technique based on the mechanisms of natural selection. In recent years, GAs have been recognized as effective and efficient techniques to solve optimization problems compared with other optimization techniques because of not requiring to take objective function derivatives and can thus deal with discrete variables and non-continuous objective functions [5,12]. Basically, GAs consists of three main stages: Selection, Crossover and Mutation. Selection directs the search of GAs toward the best individual. In the process, strings with high fitness receive multiple copies in the next generation while strings with low fitness receive fewer copies or even none at all. Crossover can cause to exchange the property of any two chromosomes via random decision in the mating pool and provide a mechanism to product and match the desirable qualities through the crossover. Mutation is a form of global search where the genetic information of a chromosome is randomly altered [5]. The application of these three basic operations allows the creation of new individuals which may be better than their parents. This algorithm is repeated for many generations and finally stops when reaching individuals that represent the optimum solution to the problem [13].

A. Chromosome Structure and Evaluation Criterion. GAs begin to explore the search space by defining a population of strings or chromosomes. Hence, first of all, the controller parameters, which will be optimized, should be coded into a string-called chromosome. In classical GAs, binary numbers are mainly used [14]. However, in this study, real number coding method is chosen in order not to consider how many bits are necessary to accurately represent a coefficient and also to reduce the processing time. The appearance of a chromosome in the population is given in Fig. 5. The chromosome consists of two genes representing the controller parameters K_p and K_i, respectively. The solution space for the proportional gain K_p^* is set to the values between 0-40 A/rad/s and the solution space for the integral gain, K_i^* is at the range 0-10 A/rad.
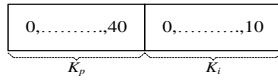
Fig. 5. Chromosome structure

The fitness of each of the chromosomes in the population is calculated using a "fitness function" that characterizes how well each particular member solves the given problem [6]. In this study, integral of the square of the error (ISE) is chosen as a fitness function

$$ISE = \int_0^k e^2 dt, \qquad (11)$$

where e represents the errors between the speed reference and the actual output speed and k is the end time for off-line iterative computation. The ultimate goal of the use of the genetic algorithm is to achieve a minimal value of the objective function so as to improve the transient and stable response quality [15].

Application of GA for PI Controller Design

During the computer simulations, the PI speed loop is reconstructed using the GA in order to achieve the desirable closed loop characteristics of the system. In this study, the genetic population consists of 8 chromosomes and the algorithm is terminated after 75 generations. The initial population is generated randomly. In each generation, a total of eight controllers (K_p and K_i pairs) are separately inserted into the DC motor drive system and the fitness values of the corresponding controllers are computed using the performance criteria defined in (11). According to these fitness values, the chromosomes in the population are sorted from the fittest to the worst. To use a 50% discard rate results in discarding the bottom four chromosomes. Then, the four with the highest fitness survive to the next generation and become potential parents to generate child population. These potential parents are selected through tournament selection and four offsprings are produced to replace the discarded chromosomes in the population. After the natural selection process, the crossover and mutation operations are performed. The proposed PI tuning based on GA is schematically shown in Fig. 6. The genetic algorithm parameters used in the simulations are listed in Table 3.
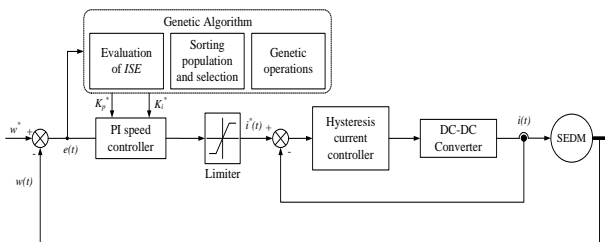
Fig. 6. Block diagram of learning the GA based optimum speed controller for DC motor control drive.

Table 3. Genetic Algorithm Parameters

Genetic algorithm parameters	
Parameters	**Values**
Population size	8

Genetic algorithm parameters	
Parameters	**Values**
Maximum generation	100
Selection method	Tournament selection
Crossover type	Uniform
Mutation rate (p_m)	0.3
Discard rate (p_d)	0.5
Elitism	On
Fitness function	*ISE*

Simulation results

In this study, the gains of a PI speed controller are adjusted employing the off-line iterative GA. Thus, GA determines the controller gains which are the most compatible and provide optimum performance. The results are obtained for cases such as step speed reference 100 rad/s and 50 rad/s under 2 Nm load. Besides, in certain intervals, motor parameters (B_m, J_m) are changed and speed responses of all the controllers are observed. A hysteresis bandwidth of 0.3 is chosen. The results of ω response and the armature current waveform relating to the genetic PI controller are given in Fig. 7, when there is a change in reference speed. At $t = 0$, reference speed is set to 100 rad/s and $t = 1$ sec, the reference speed is then decreased to 50 rad/s. In Fig. 7.a, the motor speed follows the reference speed with no overshoot. The error between the reference and the actual motor speed in the steady-state is about 0.0183 and its settling time to the reference speeds is approximately 0.36 sec and 0.115 sec, respectively. Fig. 7.b shows the actual motor current following the reference current waveform in the hysteresis band. During the transient periods, current is maintained at 50 A according to the current limiter set and the average current being injected in the steady-state stabilizes at 1.2 A.
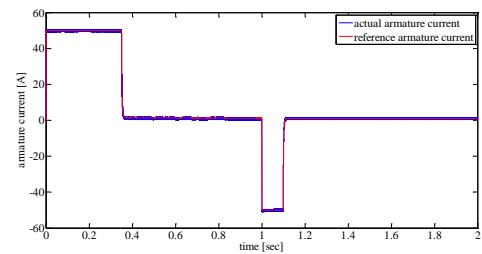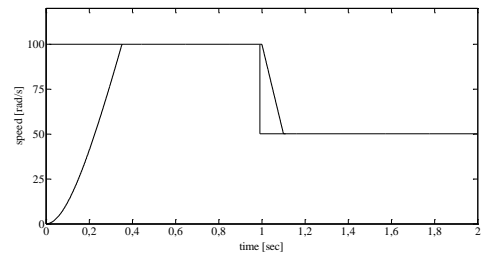
a)

b)

Fig. 7. Simulation results for the genetic PI controller under variable speed reference condition (a) speed response (b) reference and actual motor currents.

Fig. 8 shows the performances for the two controllers under variable load condition while the motor is running at 100 rad/s. At first, 10 Nm load is applied to the motor. At $t = 1$ sec, the load is completely removed from the motor

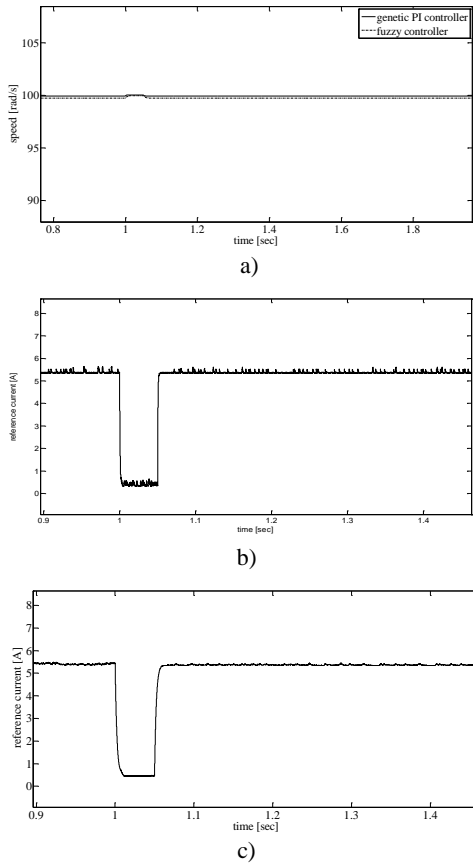shaft and at $t = 1.05$ sec, 10 Nm load is applied to the motor again.

a)

b)

c)

Fig. 8. Simulation results for the two controllers under variable load condition (a) comparative speed responses. Reference currents $i^*(t)$ using (b) fuzzy controller (c) genetic PI controller

The speed responses are seen in Fig. 8.a. It is clear that the speed response for the genetic PI controller is less affected by the load change and has a smaller steady-state error compared to the fuzzy controller.

Fig. 8.b represents the reference currents. Notice that as the reference currents for the two controllers changes so rapidly in response to the disturbances, recover time of the speed responses under the variable load condition is smaller.
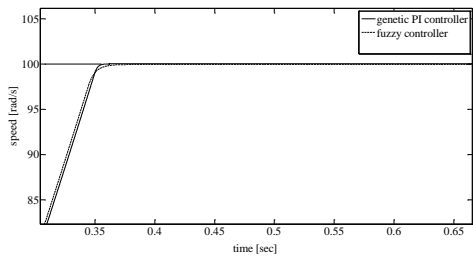

Fig. 9. Speed responses for genetic PI and fuzzy controller

A comparison for ω responses have been made between the controllers. The responses of the drive system are obtained by setting the reference speed to step 100 rad/s at $t = 0$. The comparative system responses of the controllers are shown in Fig. 9.

In Fig. 10, the performances of the genetic PI and fuzzy controller are examined by changing the motor parameters while the motor is running under 1.5 Nm. In the simulation, at $t = 1$ sec, the motor parameters B_m and J_m are increased 30 and 5 times, respectively. The motor is run for 0.5 sec under this condition and $t = 1.5$ sec, the parameters are returned to their previous values. The speed responses are seen in Fig. 10.a. During the variation of parameters, the fuzzy controller has led to a larger steady-state error than the genetic PI controller. Hence, it can be easily said that the genetic PI controller is less affected by the parameter changes compared to the fuzzy controller. Fig. 10.b shows the motor armature current for the genetic PI controller. In this condition, the motor has drawn more current to meet the increased torque, which is closer to its rated current.
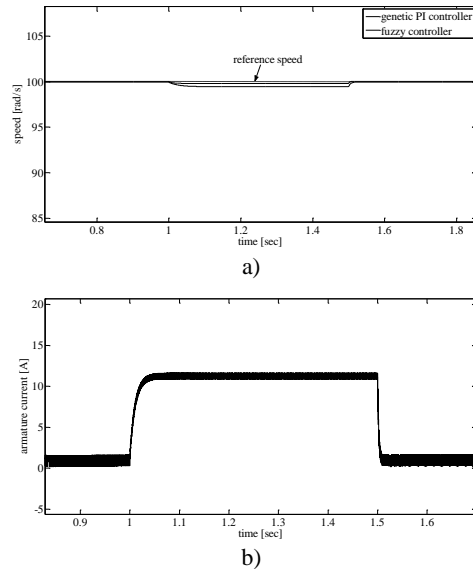

a)

b)

Fig. 10. Simulation results for the genetic PI and fuzzy controller under variable parameter condition (a) speed responses (b) motor armature current for the genetic PI controller

Fig. 11 presents distributing situation of the best controller gains in the population during the evolution procedure according to versus generation. Generation 1 shows the random initial gains. At the end of the learning algorithm, best controller gains have been found approximately K_p^*: 39.1 A/rad/s and K_i^*: 0.19 A/rad.
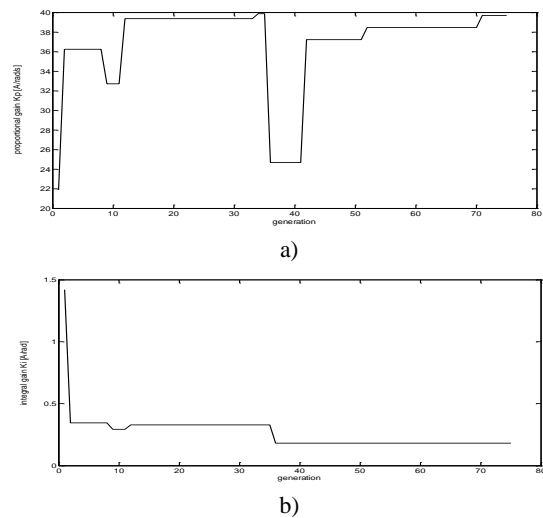

a)

b)

Fig. 11. The gain solutions of the best controller according to versus generation (a) proportional gain K_p (b) integral gain K_i

47

Conclusions

The aim of this study is to compare the two controllers namely, Fuzzy PI and Genetic PI controller for the speed controller of a separately excited motor. The control system includes a hysteresis current controller as well as speed controller. The results of application of genetic PI controller are compared to those obtained by the fuzzy PI controller under variable reference speed, variable load and variable parameter conditions and are summarized in Table 4. The paper compares the performance of fuzzy PI controller with that of genetic PI controller and it is shown that the genetic PI controller provides slightly better results in terms of the settling time and steady-state error. As a result, tuning the controller gains using genetic based algorithms will provide better dynamic response for the speed control of dc motor drives.

Table 4. Comparison of results

Specifications / Controllers	Transient performance	Steady-state performance	Complexity	Computational time
Fuzzy PI	Good	Good	Moderate	Medium
Genetic PI	Better	Better	Complicated	Long

References

1. **Krohling R. A., Jaschek H., J. P. Rey.** Designing PI/PID Controller for a Motion Control System Based on Genetic Algorithm // Proceedings of the 1997 IEEE International Symposium on. – Intelligent Control, 1997. – P. 125–130.
2. **Zadeh L. A.** Fuzzy Algorithms // Information and Control, 1968. – Vol. 12. – P. 94–102.
3. **Yu K. C., Hsu S. P., Hung Y. H**. Optimization of Fuzzy Controller of Permanent Magnet Synchronous Motor // Journal of Applied Sciences, 2007. – Vol. 7. – No. 19. – P. 2725–2735.
4. **Dubois D., Prade H.** Fuzzy Sets and Systems, Theory and Applications. – Academic Press, New York, 1980.
5. **Haupt R. L., Haupt S. E.** Practical Genetic Algorithms, 2nd ed. – Wiley–Interscience Pub., John Wiley & Sons, INC., New York, 2004.
6. **Elmas Ç., Yiğit T.** Genetic Algorithm Based On–line Tuning of a PI Controller for a Switched Reluctance Motor Drive // Electric Power Components and Systems, 2007. – Vol. 35. – No. 6. – P. 675–691.
7. **Chapman S. J.** Electric Machinery Fundamentals. – McGraw Hill, New York, 2005.
8. **Oh S. K., Jang H. J., Pedrycz W.** Optimized fuzzy PD cascade controller: A comparative analysis and design // Simulation Modelling Practice and Theory, 2011. – Vol. 19. – P.181–195.
9. **Sivanandam S.N., Sumathi S., Deepa S.N.** Introduction to Fuzzy Logic using MATLAB. – Springer–Verlag, Berlin, Heidelberg, 2007. – P. 65–81. DOI: 10.1007/978-3-540-35781-0.
10. **Bulut M., Kurt M. B., Demirtaş M.** Application of Genetic–Fuzzy Controller to a DC Motor // Journal of Polytechnic, 2004. – Vol. 7. – No. 4. – P.277–283.
11. **George M.** Speed Control of Separately Excited DC Motor // American Journal of Applied Sciences, 2008. – Vol. 5. – No. 3. – P. 227–233.
12. **Öztürk N., Çelik E.** Application of Genetic Algorithms to Core Loss Coefficient Extraction // Progress in Electromagnetics Research M (PIER M), 2011. – Vol. 19. – P.133–146.
13. **Varsek A., Urbacic T., Filipic B.** Genetic Algorithms in Controller Design and Tuning // IEEE Trans. Sys. Man and Cyber, 1993. – Vol. 23. – No. 5. – P. 1330–1339.
14. **Üstün O., Yıldız I.** Determination of the Learning Parameters in Backpropagation Learning Algorithm by Genetic Algorithm // SDU International Technologic Science, 2009. – Vol. 1. – No. 2. – P. 61–73.
15. **Wang J., Zhao Y.** The Optimization of PI Controller Parameters Using Genetic Algorithm in the DC Speed Control System // Proceedings of the 3rd World Congress. – Intelligent Control and Automation, 2000. – P. 545–548.

N. Ozturk. Speed Control for DC Motor Drive based on Fuzzy and Genetic PI Controller – A Comparative Study // Electronics and Electrical Engineering. – Kaunas: Technologija, 2012. – No. 7(123). – P. 43–48.

The aim of this paper is to design a speed controller of a DC motor by selection of a PI parameters using genetic algorithm and also to compare it with a fuzzy logic controller. The conventional PI controller has some disadvantage such as adjusting the gains, sensitivity to the controller gains and load disturbances. In order to overcome the disadvantages of the conventional PI controller, a genetic PI speed controller and a fuzzy controller for the drive system are applied to the speed loop of the DC motor and compared to each other. In the control circuits, the current control loop is also used in addition to the speed controller. Thus, the motor current is forced to stay within the hysteresis band determined by the upper and lower hysteresis limits. Simulations are realized and analyzed by the two different control strategies. It is observed that the genetic PI controller slightly enhances the performance and dynamics of the DC motor in comparison to a well-tuned fuzzy controller. The genetic PI controller is also less sensitive to the parameter variations and disturbances than the fuzzy controller. Ill. 11, bibl. 15, tabl. 4 (in English; abstracts in English and Lithuanian).

N. Ozturk. Nuolatinės srovės variklio greičio valdymas naudojant neraiškiosios logikos ir genetinį PI valdiklį – lyginamoji analizė // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2012. – Nr. 7(123). – P. 43–48.

Šio darbo tikslas suprojektuoti nuolatinės srovės variklio greičio valdiklį parenkant PI parametrus naudojant genetinį algoritmą ir palyginti jį su neraiškiosios logikos valdikliu. Tradicinis PI valdiklis turi keletą trūkumų, tokių kaip stiprinimo reguliavimas, jautrumas valdiklio stiprinimui ir apkrovos nevienodumui. Siekiant pašalinti šiuos tradicinio PI valdiklio trūkumus, genetinis PI greičio valdiklis ir neraiškiosios logikos valdiklis buvo pritaikyti nuolatinės srovės variklio greičio kontūrui ir palyginti vienas su kitu. Valdymo grandinėse srovės valdymo kilpa naudojama kaip greičio valdiklio priedas. Todėl variklio srovė verčiama likti histerezės juostoje, apribotoje iš viršaus ir iš apačios. Modeliavimas atliktas ir analizuotas taikant dvi skirtingas strategijas. Pastebėta, kad genetinis PI valdiklis truputį pagerina nuolatinės srovės variklio našumą ir dinamiką, palyginant su gerai suderintu neraiškiosios logikos valdikliu. Genetinis PI valdiklis yra ne toks mažiau jautrus parametrų kitimui ir trikdžiams kaip neraiškiosios logikos valdiklis. Il. 11, bibl. 15, lent. 4 (anglų kalba; santraukos anglų ir lietuvių k.).