Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 34 (2008) 2714–2720
Expert Systems with Applications
www.elsevier.com/locate/eswa

Optimal tuning of PI coefficients by using fuzzy-genetic for V/f controlled induction motor

Seydi Vakkas Ustun [a,1], Metin Demirtas [b,*]

[a] *Electric Education Department, Afyon Kocatepe University, Afyonkarahisar, Turkey*
[b] *Electrical and Electronics Engineering Department, Balikesir University, Balikesir, Turkey*

Abstract

This paper presents a novel speed control scheme of an induction motor using genetic-fuzzy logic. The aim of this paper is to improve a new method of the optimal tuning of proportional integral controller coefficients in the off-line control of a induction motor.

The V/f control, which realizes a low cost and simple design, is advantageous in the middle to high-speed range. Its torque response depends on the electrical time constant of the motor and adjustments of the control parameters are not need. Therefore, V/f control of induction motor is carried out. Space vector pulse width modulation with V/f is used for controlling the motor. Because, it includes minimum harmonics according to the other PWM techniques. In this paper, the first step is the identification of the system via fuzzy logic, using performance value (1/(1 + maximum overshoot and settling time)) obtained from the application circuit for different K_p–K_i pairs. In the second step, the purpose is to find the optimum controller coefficients using the fuzzy model as the objective function via genetic algorithms. A digital signal processor controller (dsPIC30F6010) was used to carry out control applications. Then, the proposed method is compared with Ziegler–Nichols method.

Keywords: Fuzzy; Induction motor; Genetic; DSP

1. Introduction

The benefits of squirrel-cage induction motors-high robustness and low maintenance make it widely used through various industrial modern processes, with growing economical and performing demands.

The V/f control, which realizes a low cost and simple design, is advantageous in the middle to high-speed range. Its torque response depends on the electrical time constant of the motor and adjustments of the control parameters are not need. V/f control is the best choice for simple variable speed applications like fans, pumps and it is control more effective in the high-speed range (Itoh, Nomura, & Ohsawa, 2002).

The motor control issues are traditionally handled by fixed gain proportional integral (PI) and proportional integral derivative (PID) controllers. However, the fixed gain controllers are very sensitivity to parameter variations, load disturbances, etc. So, the controller parameters have to be continually adapted. The problem can be solved by several adaptive control techniques such as model reference adaptive control (Sugimoto & Tamai, 1987), sliding mode control (Won & Bose, 1992), variable structure control (Chem & Wu, 1991) and self tuning PI controllers (Hung, 1994), etc. The design of all of the above controllers depends on the exact system mathematical model. However, it is often difficult to develop an accurate system mathematical model due to unknown load variation, unknown and unavoidable parameter variations due to saturation, temperature variations and system disturbances (Uddin, Radwan, & Rahman, 1987).

In high performance applications, it is useful automatically extract the complex relations that represent the drive

* Corresponding author. Tel.: +90 266 6121194; fax: +90 266 6121257.
E-mail addresses: svustun@aku.edu.tr (S.V. Ustun), mdtas@balikesir.edu.tr (M. Demirtas).
[1] Tel.: +90 272 2281311 363; fax: +90 272 2281319.

0957-4174/$ - see front matter
doi:10.1016/j.eswa.2007.05.029

behavior. The use of learning through example algorithms can be a powerful toll for automatic modeling variable speed drives (Maia, Branco, & Dente, 1994). They can automatically extract a functional relationship representative of the drive behavior. These methods present some advantages over the classical ones since they do not rely on the precise knowledge of mathematical models and parameters. On the other hand, electromechanical systems usually present internal nonlinearities and parameter deviation, which are difficult to model (Cardoso, Martins, & Pires, 1998).

PI controller is unquestionably the most commonly used control algorithm the process control industry (Yamamoto & Hashimoto, 1991). The main reason is its relatively simple structure, which can be easily understood and implemented in practice, and that many sophisticated control strategies, such as model predictive control, are based on it. In spite of its wide spread use there exists no generally accepted design method for the controller (Wang & Shao, 2000).

Most industrial processes exhibit nonlinear dynamics, and this places additional complexity on the modeling procedure used. In practice, many nonlinear processes are approximated by reduced order models, possibly linear, which are clearly related to the underlying process characteristics. However, these models may only be valid within certain specific operating ranges. When operating conditions change, a different model may be required to be used or the model parameters may need to be adapted.

System model is necessary for tuning controller coefficients in an appropriate manner (e.g., percent overshoot, settling time). Because of neglecting some parameters, the mathematical model cannot represent the physical system exactly in most applications. That's why, controller coefficients cannot be tuned appropriately.

Many of the recent developed computer control techniques are grouped into a research area called Intelligent Control, that result from the integration of fuzzy-logic techniques within automatic control systems.

The tuning of electric drive controller is a complex problem due to the many non-linearities of the machines, power converter and controller. Therefore, system model is obtained by using the fuzzy logic. The fuzzy logic is explained the next section.

2. Fuzzy logic

There is a currently a significant and growing interest in the application of artificial intelligence (AI) type models to the problem of modeling the dynamics of complex, nonlinear processes. By far the most popular type of AI model for these purposes has been the neural network, which attempts to produce 'intelligent' behavior by recreating the hardware involved in the thinking process. Another type of AI model is the fuzzy model, which defines its inputs and outputs as qualitative values (actually fuzzy reference sets) and then defines the strength of the relationships between these input and output reference sets (Saleem & Poslethwaite, 1994).

The big disadvantage of rule-based systems for dynamic modeling purposes is that the set of rules have to be formulated by one or more experts on the process behavior. The procedure which has to be gone through to obtain and rationalize these rules is rather complicated, time-consuming, and, since it involves several people with knowledge at a high technical level, rather expensive.

Unlike analytical models the fuzzy-logic model is simple, and hence computationally efficient, and at the same time, as will be illustrated for the induction motor. The fuzzy-logic model can represent complex phenomena of the system behavior more precisely. Moreover, since the model is obtained directly from the input–output data, there is no need to identify the internal system parameters in order to construct the model (Emami, Goldenberg, & Turksen, 2000).

2.1. Fuzzy variables

To obtain good model results, it is necessary to use he appropriate number of fuzzy variables and to formulate the appropriate model rules. In this study, we use the fundamental seven kinds of fuzzy variables as follows:

NL: Negative large
NM: Negative medium
NS: Negative small
ZE: Approximately zero
PL: Positive large
PM: Positive medium
PM: Positive small

The model rules for the fuzzy logic can be described by language using the input variables K_p and K_i, and the output variable, $1/(M_0 + T_s + 1)$. The ith model rule can be usually written as

Rule i: if K_p is F_i and K_i is G_i then $1/(M_0 + T_s + 1)$ is H_i

where F_i, G_i, and H_i are fuzzy variables.

In general, it is difficult to formulate control rules for unknown system. However, We already know the system and can predict a step response of the motor speed. Therefore, it is comparatively easy to formulate model rules.

To formulate model rules, it is necessary to examine the condition at each characteristic point and to consider the relation among K_p, K_i, and $1/(M_0 + T_s + 1)$ so as to bring the step response close to the set speed value (Mazaki & Sugeno, 1984; Miki, Nagai, Nishiyama, & Yamada, 1991). Finally, we can formulate model rules as shown in Table 1. Obviously from this table, fuzzy-logic model is composed of 29 control rules.

The fuzzy inference performs an important role in the fuzzy control, and the inference method used is basic and simple. As written previously, the model rules are described as follows:

Table 1
Model rules

K_i	K_p						
	NL	NM	NS	ZE	PS	PM	PL
NL		NL	NS		NM	NL	NL
NM	NL	PL	NS		NM		NL
NS		PL	NS		NM	NL	NL
ZE		PL	NS		NM	NL	NL
PS			NS				
PM	NL	PL			NM		NL
PL		NL			NM	NL	NL

Rule i: if K_p is F_i and K_i is G_i then $1/(M_0 + T_s + 1)$ is H_i

$k_p \in K_p,\ k_i \in K_i,\ 1/(m_0 + t_s + 1) \in 1/(M_0 + T_s + 1),\ i = 1, 2, \ldots, 29$

where k_{p} and k_i are numerical values of input variables and $1/(m_0 + t_s + 1)$ is the numerical values of an output variables. Fuzzy relation, R, is formed by the union of all rules as follows:

$$R = \bigcup_{i=1}^{29}$$

If the model conditions, F_0 and G_0, are given as inputs, the model output, H_0, can be obtained by $H_0 = R_0(F_0 x G_0)$. Fuzzy model membership functions are given in Fig. 1.

Many defuzzifiers have been presented in fuzzy-logic literature (Mazaki & Sugeno, 1984); however, there is no scientific or mathematical base for the preference of any of them. Consequently, defuzzificiation is considered as an art rather than simplicity. The most popular defuzzification method is the centroid method where

$$Y_{\mathrm{mean}} = \left\lfloor \int Y_{\mu B}(y)\,\mathrm{d}y \Big/ \int \mu_B(yi)\,\mathrm{d}y \right\rfloor$$

The centroid defuzzifier can be interpreted as a conditional expectation in probability distribution. However it since singleton output sets are used, a very simple defuzzification using the computed average moment is used (Mohamed & Hew, 2000). In this work, the centroid method was used.

3. Experimental setup

The experimental setup consisted of a motor and generator that was connected to it by a connecting element. The motor used was a 1.5 kW, 3.8 A, 50 Hz, $\cos\varphi = 0.82$, three phase squirrel-cage induction motor. The processor used in this work was a 10 MHz dsPIC30F6010 digital signal processor controller (DSP). The processor communicated with the PC via USB port. The block diagram of this application circuit is shown in Fig. 2. The stator voltage and frequency were adjusted using a Space Vector PWM (SVPWM) technique.

Error is calculated from difference between reference speed and actual speed taken from incremental encoder. Then, PI generates new control data according to this error. Amplitude and speed values are generated using the control data compared with V/f rate. Required values for PWM output of the DSP controller are calculated by using two values (amplitude and speed) and SVPWM technique. PWM time base is 125 μs for this application. The control loop is carried out once during each ten PWM time base. Dead time is formed by the controller. The value of dead time determined by a register is taken 7 μs.

The DSP controller program for the control process was written in dsPIC30F6010 assembly language and C30 language. Controlling and compiling process were performed by a compiler program. The experimental setup is shown in Fig. 3.

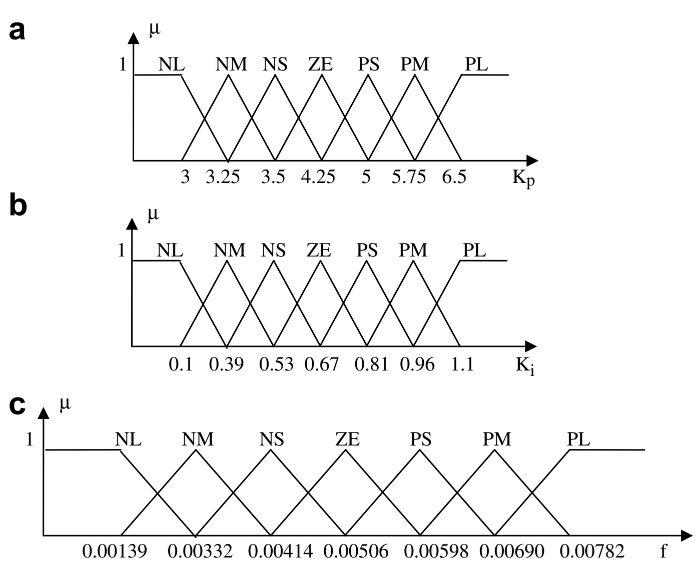

Fig. 1. Shape of membership functions for fuzzy model: (a) first input, (b) second input and (c) output.

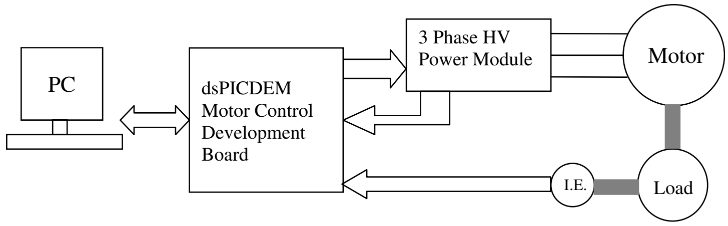

Fig. 2. The block diagram of the application circuit.

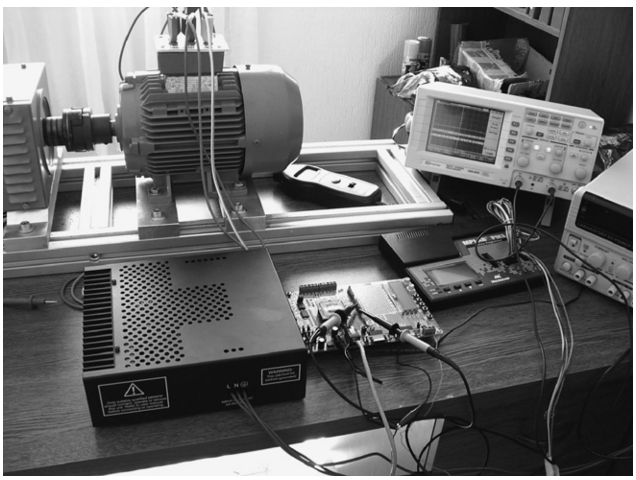

Fig. 3. The experimental setup.

4. Modeling of the induction motor using the fuzzy logic

Fuzzy logic is recently finding increasing applications that include management, economics, medicine and recently in closed loop operation of variable speed drives. The objective of the fuzzy control is to design a system with acceptable performance characteristics over a wide range of uncertainty (Miki, Nagai, Nishiyama, & Yamada, 1992; Sing, Swamy, Singh, Chadra, & Al-Haddad, 1995). The fuzzy control is basically nonlinear and adoptive in nature, giving robust performance in the face of parameter variation and load disturbance effects. Many researches (Cerruto, Consoli, Raciti, & Testo, 1997; Miki et al., 1992; Sing et al., 1995) have reported that the fuzzy-logic control yields results which are superior to those obtained using conventional control algorithms.

Fuzzy model show great potential for modeling poorly understood and highly nonlinear systems. Fuzzy models attempt to capture relationships between qualitative states and therefore represent the type of qualitative models used in everyday commonsense reasoning.

The control algorithm is based on the model of induction motor. The distinct advantage of the proposed method lies in its insensitivity to motor parameter variations.

Fuzzy sets provide an appropriate means to define operating regions. Takagi and Sugeno (1985) proposed a fuzzy modeling approach to model nonlinear systems. In their approach, the input space of a nonlinear system is divided into several fuzzy regions, and a local linear model is used in each region.

In this study, fuzzy sets are obtained using M_0 and T_s. The fuzzy rules are determined from fitness function, f, in Eq. (1)

$$f = \frac{1}{(M_0 + T_s + 1)} \tag{1}$$

The obtained value from the Eq. (1) is taken as a fuzzy output. As shown in Table 1, 29 rules are obtained using this method. Data used for fuzzy model are given in Table 2. The obtained results are shown in Fig. 4. As shown

Table 2
Data used for the fuzzy model

Data set	K_p	K_i	$f = 1/(1 + M_0\,(\text{rpm}) + T_s\,(\text{ms}))$
1	3	1	0.00177
2	3 0.3	0.00168	
3	3.25	1.1	0.00185
4	3.5	1.1	0.00185
5	3.5	0.1	0.00175
6	3.5	0.3	0.0074
7	3.5	0.5	0.00781
8	3.5	0.7	0.00752
9	3.5	0.9	0.0042
10	4.25	0.1	0.0042
.	.	.	.
.	.	.	.
28	6.5	0.9	0.00141
29	6.5	1.1	0.00139

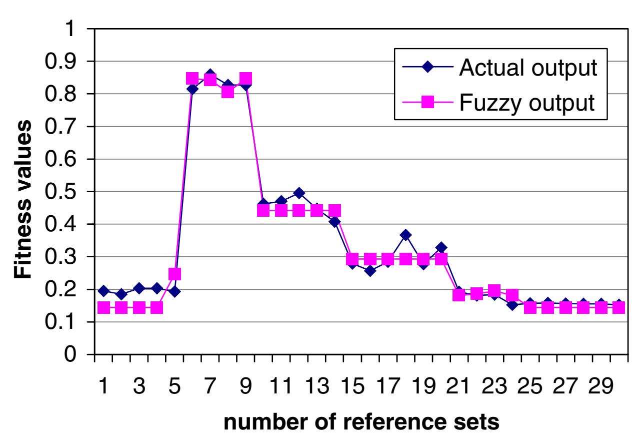

Fig. 4. The change in the real system output and fuzzy model output.

Fig. 4, there are small differences between the actual values and fuzzy output values. This also shows that the fuzzy model approach to model nonlinear system is very good.

5. Optimization of PI coefficients using GA

GAs are based on an analogy to the genetic code in our own DNA (deoxyribonucleic acid) structure, where its coded chromosome is composed of many genes (Goldberg, 1989; NgY & Li, 1995). GA approach involves a population of individuals represented by strings of characters or digits. Each string is, however, coded with a search point in the hyper search-space. From the evolutionary theory, only the most suited individuals in the population are likely to survive and generate off-spring that passes their genetic material to the next generation.

The GA is a subset of evolutionary algorithms that model biological processes to optimize highly complex cost functions. A genetic algorithm allows a population composed of many individuals to evolve under specified selection rules to a state that maximizes the "fitness" (i.e., minimizes the cost function). Some of the advantages of a genetic algorithm include that it

- Optimizes with continuous or discrete parameters,
- Does not require derivative information,
- Simultaneously searches from a wide sampling of the cost surface,
- Deals with a large number of parameters,
- Is well suited for parallel computers,
- Optimizes parameters with extremely complex cost surfaces; they can jump out of a local minimum,
- Provides a list of optimum parameters, not just a single solution,
- May encode the parameters so that the optimization is done with the encoded parameters, and
- Works with numerically generated data, experimental data, or analytical functions (Randy & Haupt, 1998).

In general GAs run repeatedly by using three basic operators such as reproduction, crossover and mutation, to find the best parameters in the whole parameter searching space. GAs are global numerical optimization methods, patterned after the natural processes of genetic recombination and evolution.

The GA used in this paper known as the simple genetic algorithm. In the algorithm, the three-operator GA with only minor deviations from the original is used (Dimeo & Lee, 1995).

Different crossover and mutation rates are used for processing of optimization of genetic algorithms. Ten of the fitness values obtained, listed from the largest fitness value to the smallest, and the fitness values of the members of the first generation are shown in Table 3. The flow chart of the GA is shown in Fig. 5 (Ustun & Demirtas, 2005).

A PI controller with the transfer function $G_c(s) = K_p + \frac{K_i}{s}$ is employed to control the process.

Table 3
Fitness values of the members, and GA parameters in the first generation

Parameter	Value
Population size	30
Crossover operator	0.90
Mutation size	0.80
Fitness of member 1	0.00751
Fitness of member 2	0.00742
Fitness of member 3	0.00721
Fitness of member 4	0.00721
Fitness of member 5	0.0068
Fitness of member 6	0.0067
Fitness of member 7	0.0060
Fitness of member 8	0.0050
Fitness of member 9	0.0050
Fitness of member 10	0.0047

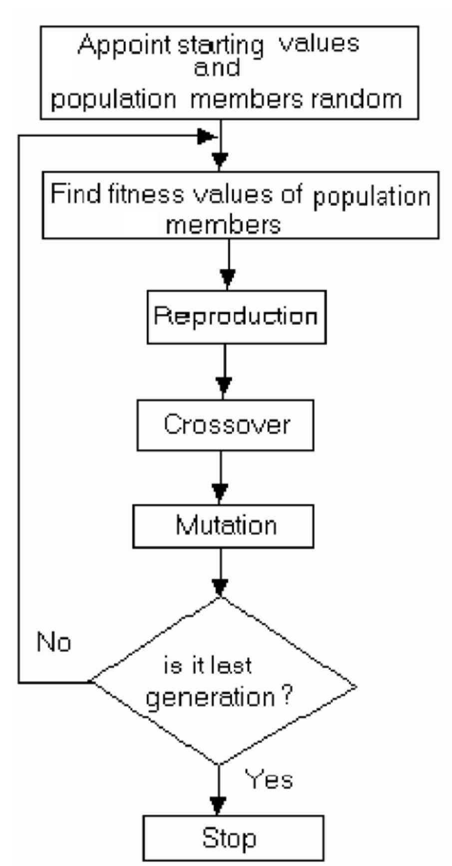

Fig. 5. The flow chart of the GA.

The optimum values for the K_p and K_i pairs were obtained using a computer program written in C++ language for the GA. This process executes with three different operators at bit level. Twenty nine of the K_p and K_i pairs were determined at random. K_p and K_i consisted of 15 bits and 12 bits, respectively. These K_p and K_i pairs were entered to fuzzy-logic model as input. The fitness values were obtained from the fuzzy output. These values were then used as the fitness function.

The one-point crossover method was used on the crossover operator. Mutual parameters of two random members on the crossover were divided into two parts and their positions were changed. A random bit of a random number on the mutation process was changed 0 to 1 and 1 to 0. For the optimization process, mutation rate is increased when converge occurs in 5–10 generation. Therefore, early converge is prevented, and in addition, members that have high-fitness values were obtained.

The range of K_p and K_i values chosen lay between (3–6.5) and (0.1–1.1) respectively. The fitness function is defined as

$$f = \frac{1}{M_0 + T_s + 1}$$

In this algorithm, the genetic algorithm parameters are selected for the training cycles were:

Population size: 30
Number of generations: 60
Crossover rate: 0.80
Mutation rate: 0.20
Chromosome length: 27 bits (15 bits for K_p and 12 bits for K_i)

6. Results and discussion

A model-based control structure is suggested that includes the fuzzy-logic dynamics model of the system. The fuzzy-logic model is systematically constructed from the input–output data.

The modeling method was tested using the induction motor data. This data consists of 29 samples of data. Each sample contains K_p, K_i inputs and M_0, T_s outputs. During this work the only the first 29 samples of data were used to identify the model. A program written in C++ language was used to generate the fuzzy model.

The optimum PI coefficients by using Ziegler–Nichols method were found to be: $K_p = 4.5$, $K_i = 0.9$. The optimum PI coefficients by using the genetic-fuzzy method were found to be: $K_p = 3.8$, $K_i = 0.6$ (generation number: 20). Optimal fitness value was not change after generation 20. Therefore, optimal K_p and K_i value are taken for generation number 20.

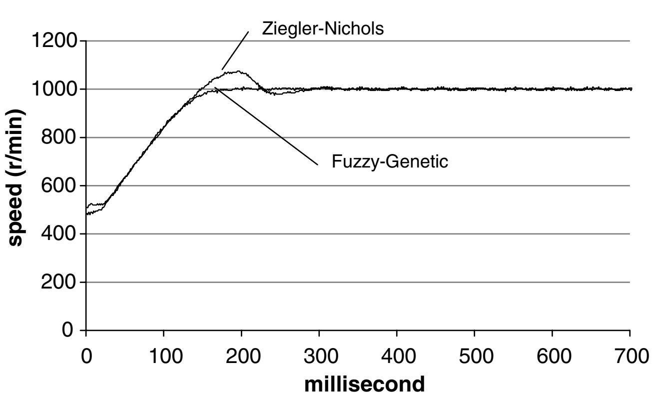

Fig. 6. The speed response.

The responses of the system for these values of K_p and K_i are shown in Fig. 6. The settling time is shorter and the maximum overshoot is minimized for these values. This shows that full system is a good control system.

The fuzzy model follows the system output, with a small error that arises from differences between experimental conditions and the model of the nonlinear system. It shows that the fuzzy model created for the system models it successfully.

The identification process is very fast and transparent, and this means that alternative model structures and reference set arrangements can be screened very quickly.

The experimental studies demonstrates the superior performance of fuzzy control, because it inherently adaptive in nature. The instant variations of the motor currents and the developed torque provide fast response of the drive system making it suitable for a number of applications such a machine tool, robotics and servo drives.

7. Conclusion

This paper describes and compares the genetic-fuzzy method with maximum efficiency and Ziegler–Nichols method. The optimal PI coefficients design method that achieves high performance for induction motor using genetic-fuzzy was proposed. Actual system (motor and controller) was modelled by fuzzy logic. It was also determined that the maximum overshoot and settling time are very small if the system is controlled by control parameters obtained from the optimization process which uses GA.

The results presented show that the fuzzy logic are able to produce accurate dynamic models of process response directly from I/O data (I: K_p–K_i, O: M_0–T_s), and GA is suitable for optimization of controller coefficients by the performance criteria considered.

This process can be also applied for nonlinear systems controlled by PD and PID controller, or a number of applications such a machine tool, robotics and servo drives.

Acknowledgements

The project is supported by TUBITAK (The Scientific & Technological Research Council of Turkey). The authors would like to acknowledge the financial support provides by TUBITAK.

References

Cardoso, F. D. S., Martins, J. F., & Pires, V. F. (1998). A comparative study of a PI, neural network and fuzzy genetic approach controllers for an AC-drive. *In IEEE, AMC'98 COIMBRA* (pp. 375–380).
Cerruto, E., Consoli, A., Raciti, A., & Testo, A. (1997). Fuzzy adaptive vector control of induction motor drives. *IEEE Transactions on Power Electronics, 12*, 1028–1039.
Chem, T. L., & Wu, Y. C. (1991). Design of integral variable structure controller and application to electrohydraulic velocity servo system. *IEE Proceedings, 138*(5), 439–444.
Dimeo, R., & Lee, K. Y. (1995). The use of a genetic algorithm in power plant control system design. *In IEEE proceeding of the 34th conference on decision* & *control* (pp. 737–742).

Emami, M. R., Goldenberg, A. A., & Turksen, I. B. (2000). Fuzzy-logic control of dynamics of systems: From modeling to design. *Engineeing Applications of Artificial Intelligence*(13), 47–69.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley.

Hung, J. C. (1994). Practical industrial control techniques. In *IEEE conference records of IECON'94* (pp. 7–14).

Itoh, J. İ., Nomura, N., & Ohsawa, H. (2002). In *A comparison between V/f control and position-sensorless vector control for the permanent magnet synchronous motor* (pp. 1310–1315). PCC-Osaka: IEEE.

Maia, J. H., Branco, P. J., & Dente, J. A. (1994). Automatic modeling of electrical drives. In *Proceedings of modern electrical drives in NATO advanced study institute* (pp. 73–78).

Mazaki, Y., & Sugeno, M. (1984). Fuzzy control. *System and Control, 28*(7), 442–446.

Miki, I., Nagai, N., Nishiyama, S., & Yamada, T. (1992). Vector control of induction motor with fuzzy pi controller. In *IEEE IAS Annual Records* (pp. 464–471).

Miki, I., Nagai, N., Nishiyama, S., & Yamada, T. (1991). In *Vector control of induction motor with fuzzy PI controller* (pp. 341–3461). IEEE.

Mohamed, H. A. F., & Hew, W. P. (2000). In *A fuzzy logic vector control of induction motor* (pp. 324–328). IEEE.

NgY, K. C., & Li, D. J. (1995). Genetic algorithms applied to fuzzy sliding mode controller design. In Murray-Smith, & K. C. Sharman (Eds.), *Proceedings of first IEE/IEEE international conference on genetic algorithms in eng. syst, innovations and applications, Sheffied, September* (pp. 220–225).

Randy, L. H., & Haupt, S. E. (1998). *Practical genetic algorithms*. A Wiley-Interscience Publication, John Wiley & Sons Inc.

Saleem, R. M., & Poslethwaite, B. F. (1994). A comparison of neural networks and fuzzy relational systems in dynamic modeling. In *Control'94, IEE* (Vol. 389, pp. 1448–1452).

Sing, B., Swamy, C. L. P., Singh, B. P., Chadra, A., & Al-Haddad, K. (1995). *Performance analysis of fuzzy logic controlled permanent magnet synchronous motor drive*. IEEE (pp. 395–405).

Sugimoto, H., & Tamai, S. (1987). Secondery resistance identification of an induction motor applied model reference adaptive system and its characteristics. *IEEE Transactions on Industry Applications, 23*(2), 296–303.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions, SMC, 15*(1), 116–132.

Uddin, M. N., Radwan, T. S., & Rahman, M. A. (1987). *Performances of novel fuzzy logic based indirect vector control for induction motor drive*. IEEE (pp. 1225–1231).

Ustun, S. V., & Demirtas, M. (2005). Optimal tuning of PI speed controller coefficients for electric drives using neural network and genetic algorithms. *Electrical Engineering, 87*(2).

Wang, Y., & Shao, H. (2000). Optimal tuning for PI controller. *Automatica, 36*, 147–152.

Won, C. y., & Bose, B. K. (1992). An induction motor servo system with improved sliding mode control. In IEEE conference records of IECON'92 (pp. 60–66).

Yamamoto, S., & Hashimoto, I. (1991). Present status and future needs: The view from Japanese industry. Chemical process control. In *Proceedings of the fourth international conference on chemical process control, TX*.