JPE 9-2-9

Improved Performance of Permanent Magnet Synchronous Motor by using Particle Swarm Optimization Techniques

A. S. Elwer* and S. A. Wahsh†

†*Electronics Research Institute, Dokki, Giza-Egypt

ABSTRACT

This paper presents a modern approach for speed control of a PMSM using the Particle Swarm Optimization (PSO) algorithm to optimize the parameters of the PI- Controller. The overall system simulated under various operating conditions and an experimental setup is prepared. The use of PSO as an optimization algorithm makes the drive robust, with faster dynamic response, higher accuracy and insensitive to load variation. Comparison between different controllers is achieved, using a PI controller which is tuned by two methods, firstly manually and secondly using the PSO technique. The system is tested under variable operating conditions. Implementation of the experimental setup is done. The simulation results show good dynamic response with fast recovery time and good agreement with experimental controller.

Keywords: PI control, Fuzzy Logic Control, Particle Swarm Optimization, Permanent magnet synchronous motor, Parameter measurement

1. Introduction

The advantages of a PMSM include high torque to current ratio, large power to weight ratio, higher efficiency and robustness. On the other hand the PMSM drive has an enlarged speed range with an inverter size lower than in conventional flux-oriented induction motor drives [1]. The intelligent PI speed controller uses the PSO algorithm to optimize the PI-parameters (Kc and τ_I) instead of the traditional trail and error method. The drive system plays an important role in meeting the other requirements. It should enable the drive to follow any reference speed tracking while taking into account the effects of load impact, saturation and parameter variation. MATLAB SIMULINK software packages are utilized to simulate each part of the system under study. The simulation of the overall system is composed of these simulated components when they are properly interconnected.

Manuscript received May 28, 2008; revised Jan. 29, 2009
†Corresponding Author: wahsh@eri.sci.eg
Tel:+20235722383, Fax:+20233351631,
*Electronics Research Institute, Dokki, Giza-Egypt

2. Methods for Tuning PI-Controllers

PI-controllers have been applied to control almost any process one could think of, from aerospace to motion control, from slow to fast systems. With changes in system dynamics and variation in operating points PI-controllers should be retuned on a regular basis. Adaptive PI-controllers avoid time-consuming manual tuning by providing optimal PI-controllers settings automatically as the system dynamics or operating points change[2]. There are various conventional methods used for tuning of PI-controller such as :

1. Trial and error
2. Continuous cycling method (Ziegler Nichols method)

3. Process Reaction Curve Methods (Ziegler-Nichols and Cohen-Coon methods)
4. Ziegler-Nichols method (both types of responses)
5. Cohen-Coon method (self regulating response only)

2.1 Trial and error [3]

PI-Controller equation is:

$$p(t) = K_c\left[e(t) + \frac{1}{\tau_I}\int_0^t e(t)dt\right] \tag{1}$$

It is quite time consuming if a large number of trial are required or if the process dynamics are slow. Testing can be expensive because of lost productivity or poor product quality

Continuous cycling may be objectionable because the process is pushed to the stability limit. Consequently, if external disturbances or a change in the process occurs during controller tuning an unstable operation or a hazardous situation could result. The tuning process is not applicable to processes that are open loop because such processes typically are unstable at high and low values of K_c but are stable at intermediate range values.

3. Particle Swarm Optimization

The PSO was originally designed by Kennedy and Eberhart[4]-[6]. The technique involves simulating social behavior among individuals (particles) “flying” through a multidimensional search space, each particle representing a single intersection of all search dimensions. The particles evaluate their positions relative to a goal (fitness) at every iteration, and particles in local neighborhood share memories to adjust their own velocities and thus subsequent positions. PSO is basically developed through simulation of bird flocking in a two-dimension space. The position of each agent is represented by its XY-axis position and the velocity is expressed by V_x (the velocity of x-axis) and V_y (the velocity of y-axis). Modification of the agent position is realized by position and velocity information. Bird flocking optimizes a certain object function. Each agent knows its best value (pbest) and its XY position. This information is an analogy of personal experiences of each agent. Moreover, each agent knows the best value [7] in the group (gbest) among bests.

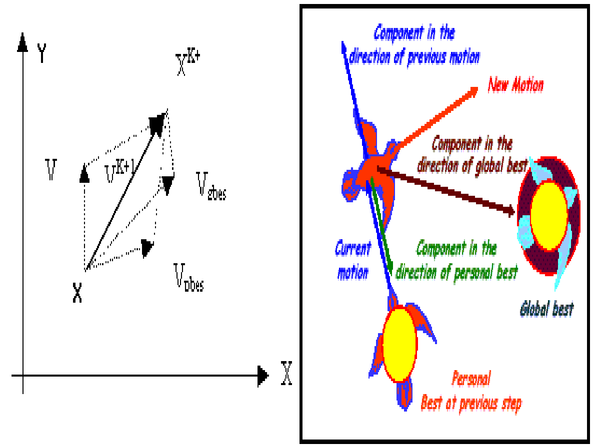


Fig. 1 Concept of modification of a searching point by PSO

This information is an analogy of knowledge of how the other agents around them have performed. Namely, each agent tries to modify its position as shown in Fig. (1).

- The current position (X,Y)
- The current velocities(V_x, V_y)
- The distance between the current position and pbest

The distance between the current position and gbest

The implementation of a PSO program is very easy and takes a few lines in the program so it reduces the whole program time. The steps of the PSO program are described in [8]-[9].

4. Modeling of PMSM in Rotor Reference Frame

The electrical system

$$v^r_q=(r_s+pL_s)i^r_{qs}+\omega_r L_s i^r_{ds}+\omega_r \lambda'^r_m \tag{2}$$

$$v^r_d=(r_s+pL_s)i^r_{ds}-\omega_r L_s i^r_{qs} \tag{3}$$

$$v_{os}=(r_s+pL_{ls})i_{os} \tag{4}$$

The expression for electromagnetic torque is obtained by the expression as follows [7]:

$$T_e=(3/2)(P/2)\lambda'^r_m i^r_{qs} \tag{5}$$

The mechanical system

$$\frac{d}{dt}\omega_r = \frac{1}{J}(T_e - B.\omega_r - T_m) \qquad (6)$$

5. Designing of PI-Controller Using PSO

The PI-controller is a good controller in the field of machine control, but the problem is the mathematical model of the plant must be known. In order to solve problems in the overall system, several methods have been introduced to tune PI-controller. Our proposed method uses the PSO to optimize the PI-controller parameters, the PSO algorithm is used on-line to update the PI-parameters (K_c and τ_I), as shown in Fig. (2)[7].

6. The Simulation Results

The overall system is simulated and tested when it is subjected to a various operating conditions. Three cases are studied to indicate the effect of increasing the number of iteration for the PSO in optimizing the PI-parameters. In the second case, the number of iteration is increased; these show that when the number of iteration increases the harmonics are decreased [8].

7. Comparison between Different Controllers

7.1 PI-Controller

A comparison between different controllers is achieved using the PI-Controller which is tuned by two methods, the first is manually and the second uses the PSO technique. The results are shown through Figure (3) to Figure (6). Figure (3a) shows the speed response using the PI-Controller that is tuned manually. From the figure it is seen that there is approximately no steady state error, but there is a high overshoot during the starting and at the loading instant. The motor response using PSO for tuning the PI-controller is shown in Figure (3b) and from the figure there is a steady state error of 20 rpm but there is no overshoot. The rising time is longer than that of Figure (3a). The speed error for both methods of tuning is shown in Figure (4). The electromagnetic torque is shown in Figure (5) and from the figure it is seen that the ripples in manual tuning are higher than those in PSO. The three-phase currents are shown in Figure (6). Note the ripples are higher when manual tuning is applied.

7.2 Fuzzy Logic Controller

The fuzzy logic controller is the second controller, which is applied to control the drive system. Also, this controller is tuned by two different methods: manual and PSO. The response of the system is shown in Figure (7) to Figure (10). The speed response of the system using FLC and tuned manually is shown in Figure (7a). From the figure it is seen that there is an overshoot and a steady state error of (-10 rpm) before loading and of (+22 rpm) after loading the motor. The speed response of the system using FLC tuned by the PSO technique, is shown in Figure (7b) which indicates that there is no steady state error before loading but has very small error after loading of (+ 2 rpm). Also there is no overshoot in the system. The speed error of the two applied methods is shown in Figure (8). The electromagnetic torque is shown in Figure (9). From the figure, one can see in the manual tuning the system is oscillated during the transient periods and this does not exist when applying PSO for the tuning purpose. The three-phase currents are shown in Figure (10). The two methods give an acceptable content of ripples.

A conclusion of the comparison between the above methods is listed in Table (1). From the table, when applying PSO in the tuning purpose it gives a good steady state error, it has no overshoot, but it takes a longer rising time.

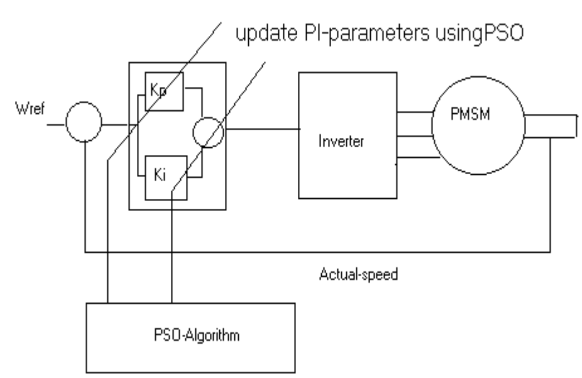
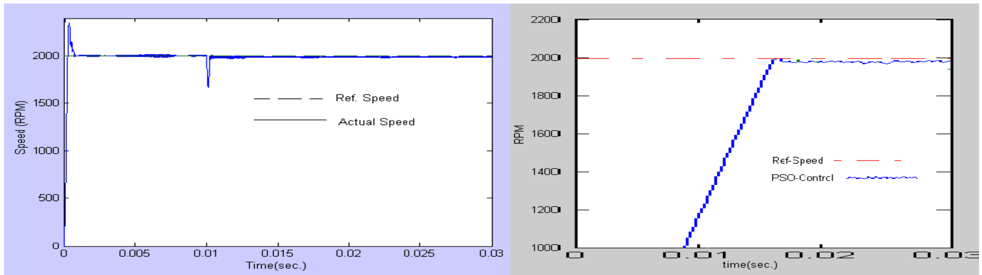
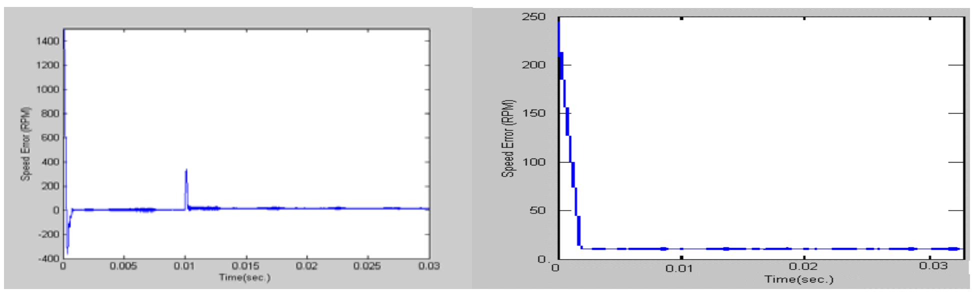


Fig. 2 Tuning PI-Controller Using PSO



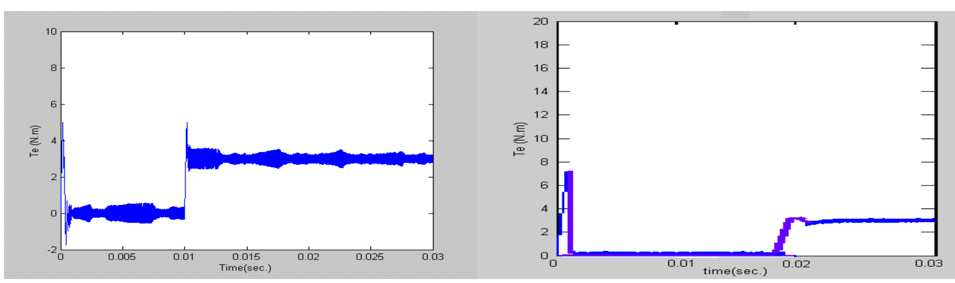
(a) Tuning PI-Controller manually (b) Tuning PI-Controller Using PSO

Fig. 3 Speed response for a step change in reference speed followed by a step change in load torque (half load)



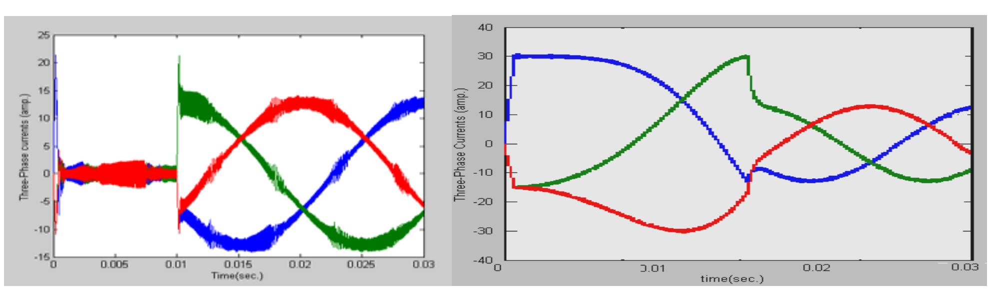
(a) Tuning PI-Controller manually (b) Tuning PI-Controller using PSO

Fig. 4 Speed error under the application of a step reference speed, followed by step reference in load torque (half load)



(a) Tuning PI-controller manually (b)Tuning PI-controller using PSO

Fig. 5 Electromagnetic torque under the application of a step reference speed, followed by step reference in load torque (half load)



(a) Tuning PI-controller manually (b) Tuning PI-controller using PSO

Fig. 6 Three-Phase currents under the application of a step reference speed, followed by step reference in load torque

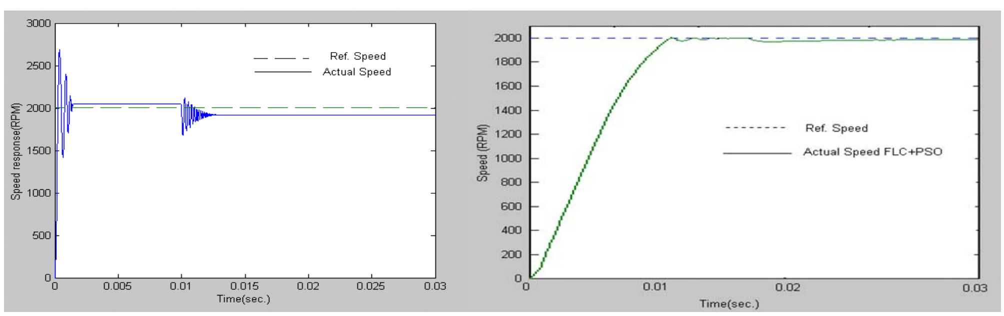
8. Experimental Setup

A laboratory setup using PMSM is prepared as shown in fig. (11). Some results such as speed response and current are shown in fig. (12) and fig. (13). A space vector modulation technique is applied to deliver an intelligent power module; the control software program is achieved using Borland C++.

A comparison between the laboratory and simulation results is shown through Figure (12) and (13). The system is tested under conditions of 1500 rpm reference speed, 5KHz switching frequency, 5,000 iteration for tuning purposes. There is a similarity between the simulation and the experimental results, so the simulated model can predict successfully the operation of the laboratory one.

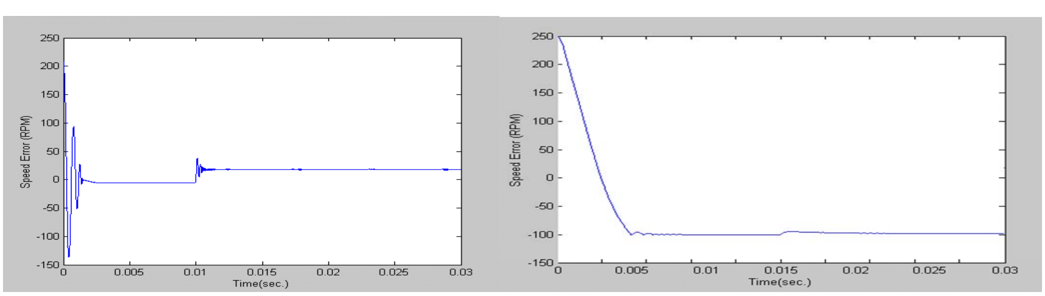
Table 1 Comparison between Different controllers

Controller / Motor response		PI tuned manually	PI tuned by PSO	FLC tuned manually	FLC tuned by PSO
Speed response	S.S. error (rpm)	0	20	-10, +22	0, +2
	Overshoot (%)	25	0	30	0
	Rising time (ms)	2	15	3	23
Electromagnetic torque (ripples)		High	Low	Medium	Medium
Three-Phase currents (ripples)		High	Low	Medium	Medium



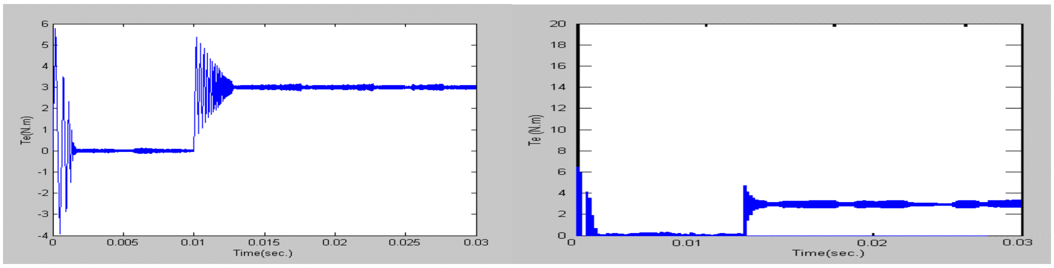
(a) Tuning FLC manually (b) Tuning FLC using PSO

Fig. 7 Speed response under the application of a step reference speed, followed by step reference in load torque (half load)



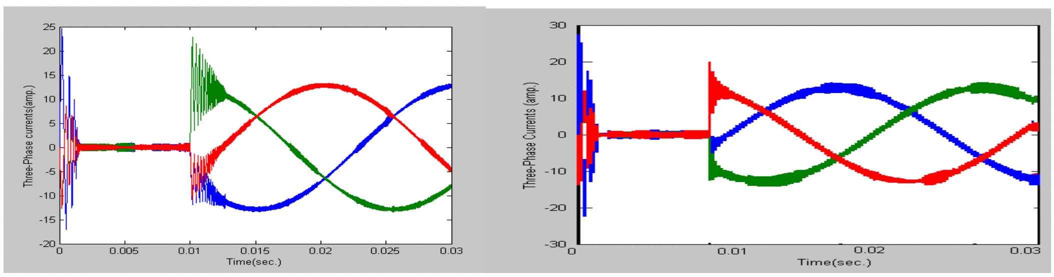
(a) Tuning FLC manually (b) Tuning FLC using PSO

Fig. 8 Speed error under the application of a step in reference speed followed by a step reference in load torque (half load)



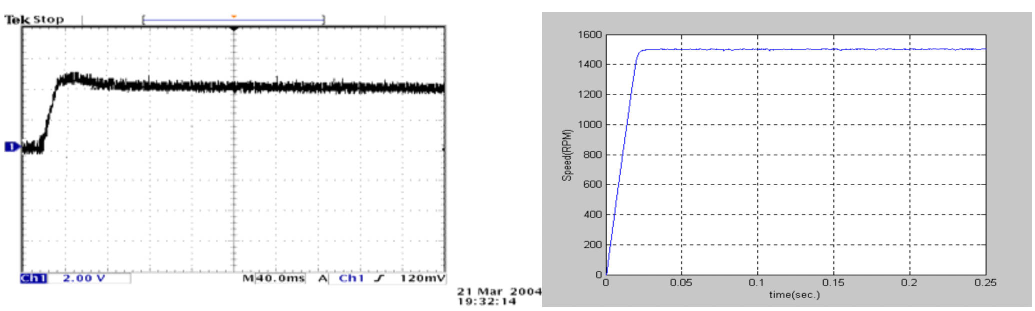
(a)Tuning FLC Manually (b) Tuning FLC using PSO

Fig. 9 The electromagnetic torque response under the application of a step in reference speed followed by a step reference in load torque (half load)



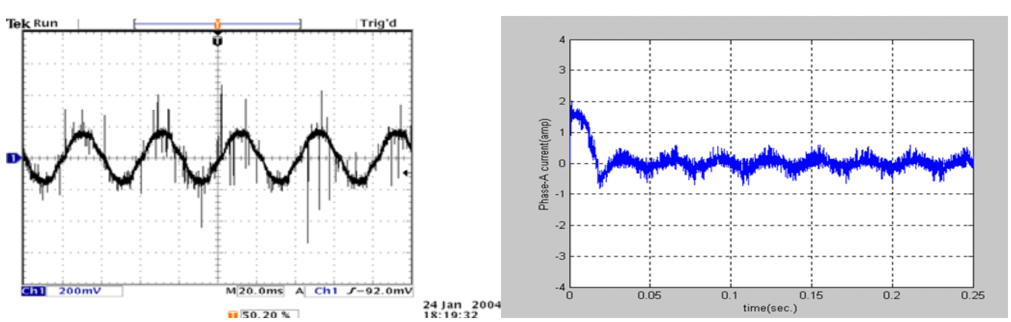
(a) Tuning FLC manually) (b)Tuning FLC using PSO

Fig. 10 Three-Phase currents under the application of a step in reference speed followed by a step reference in load torque (half load)



(a) Experimental (b) Simulated

Fig. 12 Comparison between simulation and experimental results for a step change in speed of the controlled PMSM and no load torque using PSO at 5KHz and 5,000 iteration



(a) Experimental (b) Simulated

Fig. 13 Comparison between simulation and experimental results for phase-A currents

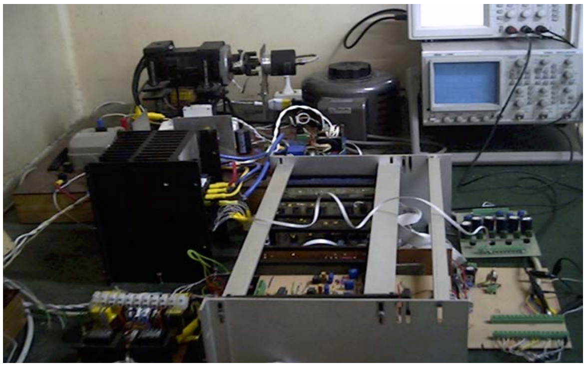


Fig. 11 The overall experimental system

9. Conclusion

The intelligent PI speed controller uses the PSO algorithm to optimize the PI-parameters instead of the traditional trial and error method. This controller is presented and tested. The Swarm Optimizer is used to adapt the PI-controller parameters. A comparison between the different controllers is achieved using the PI-Controller which is tuned by two methods, the first is manually and the second uses the PSO technique. The results show that speed response of the manually tuned PI-Controller has negligible steady state error. However there is a high overshoot during the starting and at the loading instant. The motor performance (speed/torque/current) using PSO for tuning the PI-controller has a steady state error but there is no overshoot and the rising time is longer than the previous case. The electromagnetic torque ripples in manual tuning are higher than that of the PSO. The three-phase currents have ripples higher when applying manual tuning. Implementation of the experimental setup is done. The simulation results showed good dynamic response with fast recovery time and good agreement with the experimental controller.

Appendix

The Motor Parameter

R =0.0086 ohm ,	L_{ss} =1.957mH ,
λ'^{r}_{m} =0.039 volt/rad/s	P =8
B =0.0001 (N.m)/rad/s,	J =0.00155 kg.m^2

References

[1] N Bianchi, S. Bolognani and M. Zigliotto, “High-Performance PM Synchronous Motor Drive For An Electrical Scooter”, *IEEE, Trans., on I.A.* Vol. 37, No.5, pp. 1348-1355, September/October 2001.

[2] A. Besharati, W. Lo and K. M. Tsang, “Self-Tuning of PID Controller Using Newton-Raphson Search Method”, *IEEE Trans. On I.E.*, Vol.44, No.5, pp.717-725, 1997.

[3] S. Wahsh, A., Elwer, M. Khalil, and A. Noureldin, “Tuning PI-Controller Using Particle Swarm Optimization For Control Of Permanent Magnet Synchronous Motor”, 5th St. Petersburg workshop in Simulation, 5th St. Petersburg, RUSSIA, pp.239-244, 26 June-2July 2005.

[4] R.C. Eberhart and J. Kennedy, “A New Optimizer Using Particle Swarm Theory”, in proc. of sixth International Symposium On Micro Machine and Human Science (Nagoya, Japan) , IEEE Service Center, pp. 39-43, 1995.

[5] A.P. Engelbrecht and A. Ismail, “Training Product Unit Neural Networks”, Stability and Control: Theory and Applications , Vol.2 , No.1-2 , pp. 59-74, 1999.

[6] Y.H. Shi, R. C. Eberhart, “Parameter Selection in Particle Swarm Optimization”, The 7th , Annual Conference on Evolutionary Programming , San Diego, USA. 1998.

[7] Y.H. Shi and R. C. Eberhart, “A Modified Particle Swarm Optimizer”, Proceeding of IEEE International Conference on Evolutionary Computation (ICEC’98), Anchorage, pp.69-73, May 1998.

[8] S. Wahsh, A., Elwer, M. Khalil,and A. Noureldin, “ Intelligent Fuzzy Controller Using Particle Swarm Optimization For Speed Control of Permant Magnet Motor For Electric Vehicle”, IECON’03 Conf. Proc. Roanoke, Virginia, USA, pp.1762-1766, 2-6 Nov. 2003.

[9] S. Wahsh, A., Elwer, M. Khalil, and A. Noureldin, “Intelligent PI-Controller Using Particle Swarm Optimization For Control Of Permanent Magnet Synchronous Motor For Electric Vehicle”, 7th International Symposium on Advanced Vehicle Control AVEC’04, Conf. Proc., HAN University, Arnhem, NETHERLAND, pp. 483-489, 23-27Aug. 2004.

A. S. Elwer received his B.Sc. degree in Electrical Engineering from Zagazig University, Egypt in June 1995. Then in 2001 he obtained his M.Sc from Ainshams University. Later in 2004 he got his Ph.D from Cairo University. He joined the

Electronics Research Institute (ERI) in 1997, as an assistant researcher. In 2001 he was appointed assistant professor and since 2004 he has been an associated professor. He has authored about ten articles in the field. He also works as part time in various Egyptian universities.

S. A. Wahsh received his B.Sc. degree in Electrical Engineering from AinShams University, Cairo, in June 1967. In 1973 he joined the National Research Center (NRC) as a researcher. He received his Ph.D. in Electric Drives from Budapest Technical University, Hungary in January 1980. Since 1980 he has been working as an assistant professor in the Electronics Research Institute (ERI). From 1984 till 1998 he was head of Power Electronics and Energy Conversion Dept., ERI. In 1990 he became a professor of Power Electronics in ERI. He has authored more than a hundred articles in the field of modeling, analysis and control of DC, IM, SRM and PMSM. Furthermore, he has supervised several M Sc and PhD thesis. He has also managed several projects as principal investigator. Prof. Wahsh works part time in various Egyptian universities. He was a visiting professor for various foreign universities. He has done a considerable amount of consulting engineering work in Industrial Electronics .His major field of interest includes Power Electronics, Electric Drives, Motion Control and Electrical Vehicles.