Journal of Engineering and Technology Vol. 4 No. 2, 2005 ISSN 1684-4114

DESIGN AND IMPLEMENTATION OF IPMSM DRIVES USING ARTIFICIAL INTELLIGENCE.

M. J. Hossain*, K. K. Islam** , M. A. Hoque* and M. A. Rahman***

ABSTRACT

This paper presents novel approaches of speed control scheme for interior permanent magnet synchronous motors (IPMSM) using artificial intelligence. Two types of control techniques based on fuzzy logic and neural networks are presented as examples of intelligent electrical drives. The complete vector control scheme of the IPMSM drive incorporating the artificial intelligent controllers i.e. fuzzy logic controller (FLC) and artificial neural network (ANN) are simulated and experimentally implemented for an 1-hp IPMSM using a digital signal processor board DS-1102 in the laboratory. Performance of the proposed FLC-and ANN-based IPMSM drive is investigated and compared for various dynamic operating conditions, such as, sudden change in command speed, step change in load, etc. The comparative results show that both the controllers provide optimum performances but the FLC is found to be superior over the ANN in terms of lower settling time and computational burden for the high-performance industrial drive applications.

Key Words: - Digital signal processor, fuzzy-logic controller, artificial neural network, interior permanent magnet synchronous motor, on-line system identification, and hysteresis current controller.

1. INTRODUCTION

In recent years IPMSM has become increasingly popular for its use in high performance drive (HPD) applications due to desirable features, such as, high torque to current ratio, high power to weight ratio, high efficiency, low noise and robust operation. The advantageous features of the IPMSM for modern drives application are well established [1-2]. In high-performance drive systems, the motor speed should closely follow a specified reference trajectory regardless of any load disturbances, parameter variations, and model uncertainties.

* Dept. of Electrical & Electronic Engineering, Rajshahi University of Engineering & Technology, Rajshahi-6204, Bangladesh

** Dept. of Electrical & Electronic Engineering, Islamic University of Technology, Gazipur-1704, Bangladesh

*** Faculty of Engineering & Applied Science, Memorial University of Newfoundland, St. John's, Newfoundland A1B 3X5, Canada

However, the controller design of such a system plays a crucial role in system performance. The decoupling characteristics of vector-controlled IPMSM are adversely affected by the parameter changes in the motor. 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 sensitive to parameter variations, load disturbances, sudden speed change etc. Thus, the controller parameters have to be continually adapted. The problem can be solved by several adaptive control techniques, such as, model reference adaptive control (MRAC) [2], sliding-mode control (SMC) [3], variable structure control (VSC) [4], and self-tuning PI controllers [5], 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. In order to overcome the above problems, recently, the artificial intelligence controllers are being used for motor control purpose [7]–[12]. There are different approaches in the field of artificial intelligence for realizing human intelligence in machines. In recent years, fuzzy logic controller and artificial neural network have emerged, as a practical technology with successful applications in many fields. As compared to the conventional PI, PID, and their adaptive versions, artificial intelligence has some advantages, such as: it does not need any exact system mathematical model and it can handle nonlinearity of arbitrary complexity. Neural networks and fuzzy control can be usefully employed to solve nonlinear control and modeling problems, simplifying and automatizing both the process modeling and the controller synthesis phases. The most widely used algorithm is the back propagation. The main advantage of this algorithm is its simplicity. However, the main drawback of back-propagation is the large number of iterations that are required to obtain convergence.

The main advantage of the fuzzy logic over an ANN based approach is its simplicity to be implemented. Unfortunately, with the increase in the complexity of the process being modeled, the difficulty in developing dependable fuzzy rules and membership functions increases. Hence, in this work the fuzzy logic based algorithm is simplified and incorporated in the drive system. The complete vector control scheme of IPMSM incorporating the FLC and ANN have been successfully simulated and implemented in the laboratory. The performances of the FLC based drive have also been compared with those obtained from the ANN based drive at different operating conditions. It is found that the both the proposed FLC and ANN based drives are insensitive to temperature changes, inertia variations, and load torque disturbances but the FLC based drives provides better performance in terms of lower settling time, computational burden and implementation complexity.

2. IPMSM DYNAMICS

The mathematical model of an IPMSM drive can be described by the following equations in a synchronously rotating rotor d-q reference frame as,

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} R+pL_d & -P\omega_r L_q \\ P\omega_r L_d & R+pL_q \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} 0 \\ P\omega_r \psi_f \end{bmatrix} \tag{1}$$

$$T_e = T_L + J_m p\omega_r + B_m \omega_r \tag{2}$$

$$T_e = \frac{3P}{2}\left(\psi_f i_q + (L_d - L_q) i_d i_q\right) \tag{3}$$

where,

v_d, v_q = d- and q-axis stator voltages;
i_d, i_q = d- and q-axis stator currents;
R = stator per phase resistance;
L_d, L_q = d- and q-axis stator inductances;
J_m = moment of inertia of the motor and load;
B_m = friction coefficient of the motor;
P = number of poles of the motor;
ω_r = rotor speed in angular frequency;
p = differential operator (=d/dt);
ψ_f = rotor magnetic flux linking the stator;
T_e, T_L = electromagnetic and load torques;

3. DESIGN OF SIMPLIFIED FLC FOR IPMSM

In this work, the fuzzy logic controller is based on the fuzzy set and fuzzy logic theory introduced by Zadeh [16], with the vector control techniques incorporated with the FLC to obtain the highest torque sensitivity of the IPMSM drive. The vector control technique is formulated within the d-q synchronously rotating rotor reference frame. The complexity of the control arises due to the nonlinear nature of the torque expressed by (3). Moreover, L_d and L_q undergo significant variations at different steady state and dynamic loading condition [2]. The dynamic model of the IPMSM may be rewritten from (1-3) as

$$pi_q = (v_q - Ri_q - K_b\omega_r)/L_q \tag{4}$$
$$p\omega_r = (T_e - T_L - B_m\omega_r)/J_m \tag{5}$$

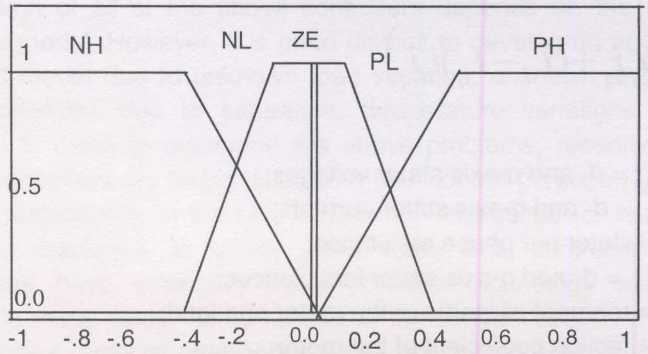
where $K_b=p\lambda_f$. As the FLC can handle any non-linearity, one can consider the load as unknown nonlinear mechanical characteristics.

The load can be modeled using the following equation as [4],

$$T_L = A\omega_r^{\ 2} + B\omega_r + C \tag{6}$$

where A, B, C are arbitrary constants. To make the control task easier, the equation of an IPMSM are expressed as a single input and single output system by combining (5) and (6) in continuous time domain form as

$$J_m \frac{d\omega_r}{dt} = T_e - (B_m + B)\omega_r - A\omega^2{}_r - C. \tag{7}$$



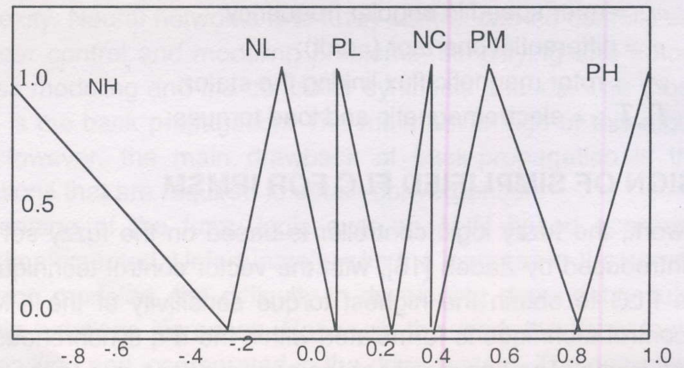


Fig.1 Member functions for: (a) Speed error $\Delta\omega_m$. (b) Command Torque T_e^*

A small incremental change ΔT_e of the electrical torque T_e results in a corresponding change $\Delta\omega_r$ of the speed ω_r. Then (7) can be rewritten as,

$$J_m \frac{d(\Delta\omega_r)}{dt} = \Delta T_e - (B_m + B)(\Delta\omega_r) - A(\Delta\omega^2{}_r). \tag{8}$$

By replacing all the continuous quantities of (8) by their finite differences, the discrete time signal model of the simplified IPMSM with nonlinear load can be given as $\Delta T_e(n) = \frac{-J_m}{t_s}\Delta e(n) + (B_m + B)\Delta\omega_r(n) + A\{\Delta\omega_r(n)\}^2$ (9)

Hence, $T_e(n) = \int_{discrete} \Delta T_e(n) = f(\Delta e(n), \Delta\omega_r(n)$ (10)

where, $\Delta e(n) = \Delta\omega_r(n) - \Delta\omega_r(n-1)$ is the change of speed error, $\Delta\omega_r(n) = \omega_r^*(n) - \omega_r(n)$ is the present sample of speed error, $\Delta\omega_r(n-1)$ is the past sample of speed error, $\omega_r(n)$ is the present sample of actual speed, $\omega_r^*(n)$ is the present sample of command speed, t_s is the sampling time interval and f denotes the nonlinear function. Thus, the purpose of using the FLC is to map the nonlinear functional relationship between electrical torque T_e and rotor speed ω_r.

From this command torque T_e, (16) are used to calculate the necessary q- axis current to produce the rotor speed ω_r. In real time the motor position information and output of the simplified FLC in terms of the command q-axis and d-axis currents i_q^* and i_q^* are used to get the motor command phase current i_a^*, i_b^* and i_c^* by using Park's transformation.

The model of the IPMSM expressed by (10) defines the input and output linguistic variables for the FLC of the IPMSM drive. According to (10), the input of the proposed FLC are the present sample of speed error and the change of speed error, which is the difference between present and past sample of speed errors. However, it has been observed that the effect of the inclusion of the change of speed error on the motor speed response is negligible, and does not produce an improvement in motor drive performance in measure with the necessary increase in computational burden as compared to when it is omitted. The omission of the $\Delta e(n)$ term produces an FLC-based drive with acceptably responsive and accurate tracking of the command speed. Thus, the input vectors of the FLC can be reduce to only $\Delta\omega_r(n)$, producing a much simplified FLC as compared to input vectors of $\Delta\omega_r(n)$ and $\Delta e(n)$ with the non-simplified system. This simplification reduces computational burden and lowers the computer power required to implement the FLC scheme in real time. Thus this simplified FLC is a significant factor for real-time implementation of the laboratory IPMSM drive system. The block diagram of the proposed controllers based IPMSM drive is shown in Fig. 2. Next, scaling factors K_w and K_I are chosen for fuzzification and obtaining the appropriate actual command current. The factor K_w is chosen so that the normalized value of speed error, $\Delta\omega_r$ remains within the limit of $\pm$ 1. The factor K_I is chosen so that the rated current i produced by the controller for rated conditions. In this paper, the constants are taken as $K_w = \omega_r^*$ (command speed) and $K_I=10$ in order to get the optimum drive performance.

After selecting the scaling factors, the next step is to choose the membership functions of $\Delta\omega_r$ and i_{qn}^*, which form an important element of the FLC. The membership functions used for the input and output fuzzy sets are shown in Fig. 1. The trapezoidal functions are used as membership functions for all the fuzzy sets except the fuzzy set ZE (zero) of the input vectors. The triangular membership functions are used for the fuzzy set ZE of the input vectors and all the fuzzy sets of the output vector. Mathematically, the trapezoidal membership function can be defined as

$$\text{Trapezoidal: } f(x; a, b, c, d) = \begin{bmatrix} 0 & x \le a \\ \frac{x - a}{b - a} & a \le x \le b \\ 1 & b \le x \le c \\ \frac{d - x}{d - c} & c \le x \le d \\ 0 & x \ge d \end{bmatrix} \tag{11}$$

The triangular membership function can be obtained from the trapezoidal function by setting b=c. The rules used for the proposed FLC algorithm are shown in Table 1. Based on the rules, the fuzzy-rule-based matrix is shown in Table 2. For this study, Mamdani-type fuzzy inference is used [16]. The values of the constants, membership functions, fuzzy sets for the input/output variables, and the rules used in this study are selected by trial and error to obtain the optimum drive performance. In this study, the center of gravity defuzzification is used. The output function is given as

$$i = \frac{\sum_{k=1}^{N} i\mu_{C(k)}(i)}{\sum_{k=1}^{N} \mu_{C(k)}(i)_d} \tag{12}$$

where N is the total number of rules and $\mu_{C(k)}(i)$ denotes the output membership grade for the kth rule with the output subset C.

Table 1

Fuzzy rules

(i) if $\Delta\omega_r$ is PH (positive high), T_e is PH (positive high).

(ii) if $\Delta\omega_r$ is PL (positive low), T_e is PM (positive medium).

(iii) if $\Delta\omega_r$ is NL (Negative low), T_e is NL (Negative low).

(iv) if $\Delta\omega_r$ is NH (Negative high), T_e is NH (Negative high).

Table 2

Fuzzy-Rule-Based Matrix

$\Delta\omega_r$	PH	PL	NL	NH	ZE
T_e	PH	PL	NL	NH	NC

Fig.2 shows the control scheme of the motor drive. The command torque is obtained from a FLC or ANN type speed controller. Using equation (16), reference q- axis current i_q^* is computed first, subsequently reference d-axis current i_d^* is also calculated. Using these reference currents, three phase currents are determined by vector rotator. The hysteresis current controller compares the reference three phase currents with actual currents and generates base signals for the transistorized inverters.

There are different types of membership functions available in fuzzy logic control, such as, trapezoidal, triangular, rectangular etc. Among them mainly triangular type membership functions are used in this paper in order to reduce the computational burden for online implementation.

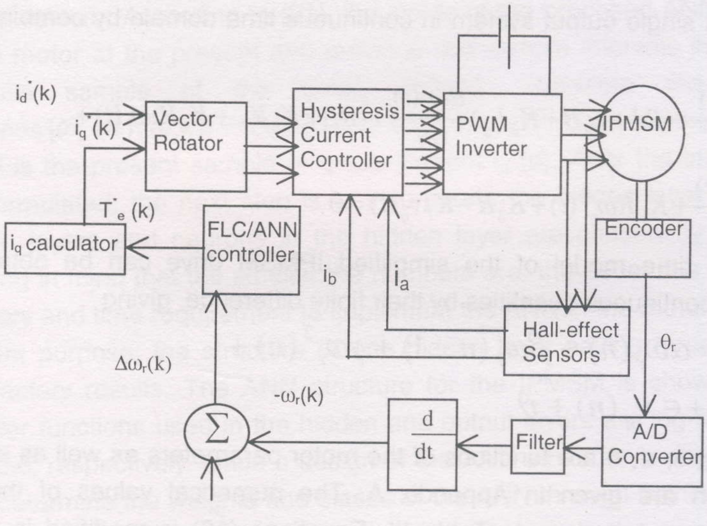


Fig.2 Block diagram for the proposed controller based IPMSM Drive.

4. ANN STRUCTURE FOR IPMSM

The developed torque in (2) can have any non-linear unknown mechanical characteristics. The following equation can be used to model a non-linear load [14].

$$T_L = K_1\omega_r^{\ 2} + K_2\omega_r + K_3 \tag{13}$$

where K_1, K_2 and K_3 are constants. An efficient control strategy of the vector control technique is to make the d-axis current i_d zero so that the direct flux linkage λ_d becomes dependent only on the flux linkage by the permanent magnet rotor. With this control strategy, the machine model becomes simpler as can be described by the following equations

$$pi_q = (v_q - Ri_q - K_b\omega_r) / L_q \tag{14}$$

$$p\omega_r = (T_e - T_L - B_m\omega_r) / J_m \tag{15}$$

where $K_b = P\lambda_f$ and the developed torque in (3) is proportional to the q-axis current as given by

$$T_e = K_T i_q \tag{16}$$

where $K_T = \frac{3}{2}P\lambda_f$.

Equation (16) resembles the torque equation of a separately excited dc motor, where i_q corresponds to the armature current of a dc machine. Hence, a precise torque control of the IPMSM is possible by controlling the q-axis current I_q. Now to make the control task easier, the equations of a IPMSM can be expressed as a single input single output system in continuous time domain by combining (13-16), giving

$$L_qJ\frac{d^2\omega_r(t)}{dt^2} + (RJ + L_qB + K_2L_q)\frac{d\omega_r(t)}{dt} + (RB + K_bK_T + K_2R)\omega_r(t) +$$

$$K_1L_q\frac{d\omega^2(t)}{dt} + K_1R\omega_r^{\ 2}(t) + K_3R - K_Tv_q(t) = 0 \tag{17}$$

The discrete time model of the simplified IPMSM drive can be obtained by replacing all continuous quantities by their finite difference, giving

$$\omega_r(n+1) = \alpha\omega_r(n) + \beta\omega_r(n-1) + \gamma\omega_r^{\ 2}(n) +$$

$$\delta\omega_r^2(n-1) + \in v_q(n) + \vartheta \tag{18}$$

where α, β, γ, δ, $\in$, ϑ are functions of the motor parameters as well as sampling interval which are given in Appendix A. The numerical values of the motor parameters are also given in Table III. Equations (18) is modified in order to obtain the inverse model of the drive system as

$$\omega_r(n+1) = [\omega_r(n+1) - \alpha\omega_r(n) - \beta\omega_r(n-1) - \gamma\omega_r^{\ 2}(n) -$$

$$\delta\omega_r^2(n-1) - \vartheta] / \in \tag{19}$$

Now in discrete form, the q-axis current can be expressed in terms of v_q (n) and ω_r (n) by replacing the continuous terms of equation (14) by their finite difference, giving

$$i_q(n) = Ai_q(n-1) + B_2 v_q(n) + C\omega_r(n) \tag{20}$$

where

$$A = 1 - \frac{R\Delta T}{L_q}, \qquad B_2 = \frac{\Delta T}{L_q}, \qquad C = -\frac{\Delta T K_b}{L_q}$$

and ΔT is the sampling interval. Thus the expressions for the q-axis current can further be modified as

$$i_q(n) = Ai_q(n-1) + B_2[\omega_r(n) - (\alpha + \frac{\in C}{B})\omega_r(n-1) -$$

$$\beta\omega_r(n-2) - \gamma\omega_r{}^2(n) - \delta\omega_r^2(n-2) - \vartheta]/\in \tag{21}$$

The right hand side of (21) is a nonlinear function of the speed ω_r. The purpose of using the ANN is to map the nonlinear relationship between the q-axis current i_q (n) and the speed ω_r (n) according to (21). Equation (21) reveals the structure of the ANN for the speed control of the IPMSM. One of the important aspects of applying an ANN to any particular problem is to formulate the inputs and output of the ANN structure under study. The inverse dynamics of the IPMSM as described in equation (21) dictates the inputs and output of the ANN used in the control system. According to (21), the inputs of the proposed ANN are the speed of the motor at the present and previous two sample intervals in addition to the previous sample of the q-axis current. Therefore the input vector becomes $[\omega_r(n), \omega_r(n-1), \omega_r(n-2), i_q(n-1)]$. The corresponding output target is the present sample of q-axis current i_q (n). After the inputs and output are formulated, the next step is to incorporate the hidden layer (s). Number of hidden layers and neurons in the hidden layer are chosen by trial and error, keeping in mind that the smaller the numbers are, the better it is in terms of both memory and time requirement to implement the ANN in the motor control. For the present purpose, the structure of one hidden layer having three neurons gives satisfactory results. The ANN structure for the IPMSM is shown in Fig.3. The transfer functions used in the hidden and output layers are log-sigmoid and tan-sigmoid, respectively. Once a design of the ANN structure is done, the next step is to determine the weights and biases of the ANN through training to achieve the specific target with the given inputs. The back-propagation-training algorithm is used for this purpose, which is based on the principle of minimization of a cost function of the error between the outputs and the target of the ANN [14].

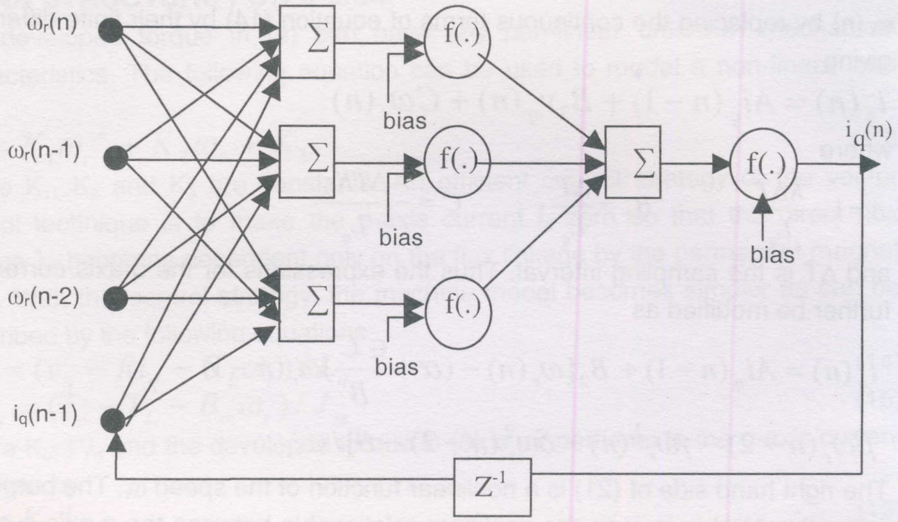


Fig.3 ANN structure for IPMSM drive

Depending on the applications, the training of the ANN could be off-line or on-line. If the weights and biases of the ANN is determined through off-line training only then an intensive training has to be performed considering all operating conditions of the system, which are almost impossible for the control of a IPMSM. As for example in reference [14], the load is modeled in (13), which is not always true in practical situations. Hence, the need for on-line weight and biases up-dating arises. However, the on-line training task could be eased and the system can be made more stable if an initial set of weights and biases is generated a priori through the off-line and on-line training has been used for the present work. The initial weights and biases are obtained through the off-line training. Training data are obtained from the simulated PI controlled drive system using simulink model [15].

The input vector was $[\omega_r(n), \omega_r(n-1), \omega_r(n-2), i_q(n-1)]$ and output vector was i_q (n). Several sets of data were obtained for different operating conditions. The designed ANN was trained with these simulated data. These are up-dated only when an error limit between the actual output and the target of the ANN exceeds a preset value.

5. LABORATORY IMPLEMENTATION

The complete IPMSM drive system as shown in Fig.4 has been implemented in the laboratory for a 1-hp laboratory IPMSM using DSPACE DSP controller board [8]. The DSP board is installed in a PC with uninterrupted communication capabilities through dual-port memory. The DSP has been supplemented by a set of on-board peripherals used in digital control systems, such as A/D, D/A converters and incremental encoder interfaces. The DS 1102 is also equipped with a TI TMS320P14, 16-bit micro controller DSP that acts as a slave processor and is used for some special purposes. In this work, slave processor is used for digital I/O configuration. The actual motor currents are measured by the Hall-effect sensors which have good frequency response and fed to the DSP board through A/D converter. As the motor neutral is isolated, only two-phase currents are fed back and the other phase current is calculated from them. Three phase reference currents are generated utilizing reference q- and d-axis currents and rotor position angle obtained through encoder mounted on the shaft of the motor. Computed three phase reference currents are converted to upper and lower hysteresis by adding and subtracting a reselected band. Hysteresis currents are compared with actual motor currents and PWM base drive signals are generated. All computations for generating reference currents and consequently base drive signals for the inverter are done by

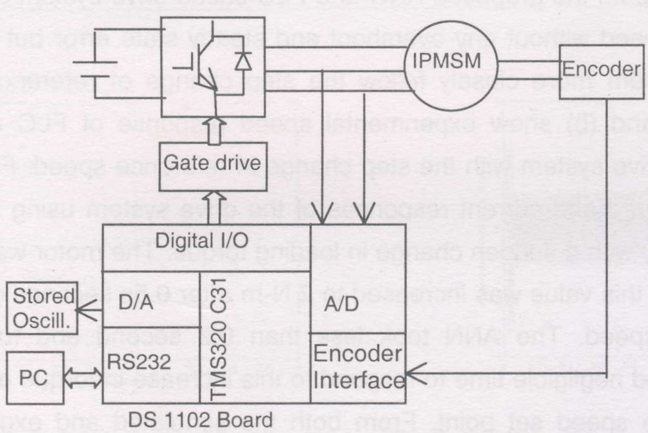


Fig.4 Hardware schematic for real-time implementation of IPMSM drive.

developing a program in ANSI C programming language. The program is compiled using Texas Instrument C compiler and downloaded to the DSP controller board. The sampling frequency for experimental implementation of the proposed drive is 10 kHz.

6. RESULTS AND DISCUSSIONS

This paper has presented and compared the performances of the fuzzy-based IPMSM drive with those obtained from ANN-based IPMSM drive. The speed and current responses are observed under different operating conditions, such as, various command speeds, sudden application of load, step change in command speed and at different loading conditions. Some of the sample results are presented in this paper. Figures 5 and 7 show the simulated starting performance of the drive with FLC and ANN-based drive systems respectively. From the Fig 5, we observed that the response of speed characteristics is faster by using FLC than that of with ANN. Figures 6 and 8 show experimental speed and steady state current response of FLC and ANN based IPMSM drive system with the reference speed of 188.5 rad/s at a load of 2 N-m. Figures 9 (a) and (b) show simulated speed responses of the drive system using ANN and FLC, respectively with a step change in the reference speed. Initially the motor speed was 160 rad/s and after 0.5s the speed is increased to 188.5 rad/s. It is evident from Fig. 9 (a) and (b) that both the proposed ANN and FLC based drive system can follow the command speed without any overshoot and steady state error but the FLC-based drive system more closely follow the step change of reference speed. Figures 10 (a) and (b) show experimental speed response of FLC and ANN based IPMSM drive system with the step change in reference speed. Figures 11 and 12 show speed and current responses of the drive system using ANN and FLC, respectively with a sudden change in loading torque. The motor was started with no load and this value was increased to 2 N-m after 0.5s seconds causing a drop in motor speed. The ANN took less than 0.2 second and fuzzy logic controller required negligible time to respond to this increase in torque and return the motor to the speed set point. From both the simulated and experimental results, it is clear that both ANN and FLC-based drives provides optimum performances, but the FLC based drive system is superior than the ANN-based drive system in terms of faster response, computational burden and implementation complexity. The comparisons of required calculations for a single sample for ANN and FLC are shown in table 3.

Table 3

Comparison of calculations between ANN and FLC required for one sample

	ANN	FLC	
Input Layer	12- Products	Normalization sections	1-Division
Hidden Layer	3-Summations 3-Bias addition 3-Operations on transfer function	Fuzzification	4-Comparisons
Output Layer	3-Products 1-Summation 1-Operation on transfer function 1-Bias addition	Defuzzification and Denormalization	4-Multiplication 2-Summation 1-Division 1-Multiplication

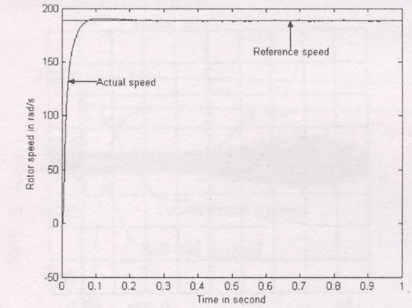


Fig.5 (a) Simulated speed of IPMSM drive at a reference speed of 188. 5 rad/s using ANN

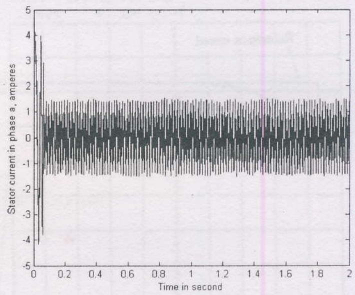


Fig.5 (b) Simulated current of IPMSM drive at a reference speed of 188. 5 rad/s using FLC

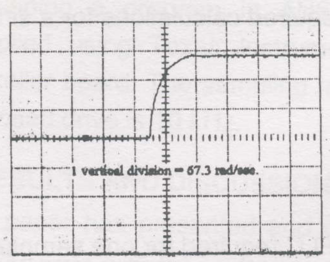


Fig.6 (a) Experimental speed of IPMSM drive at a reference speed of 188. 5 rad/s using ANN.

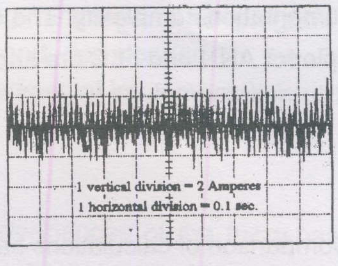


Fig.6 (b) Experimental current of IPMSM drive at a reference speed of 188. 5 rad/s using FLC.

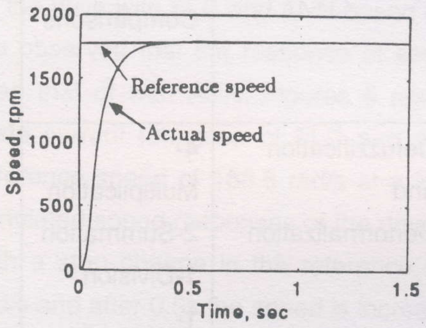


Fig.7 (a) Simulated speed of IPMSM drive at a reference speed of 188. 5 rad/s using ANN.

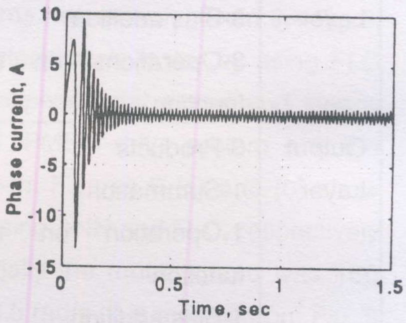


Fig.7 (b) Simulated speed of IPMSM drive at a reference speed of 188. 5 rad/s using FLC.

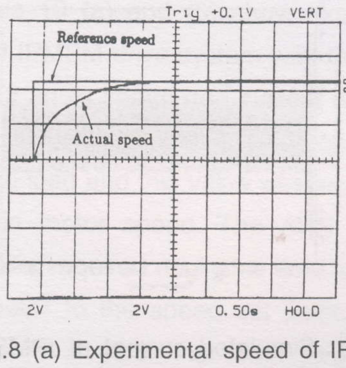


Fig.8 (a) Experimental speed of IPMSM drive at a reference speed of 188. 5 rad/s using ANN. 1 division=65 rad/s

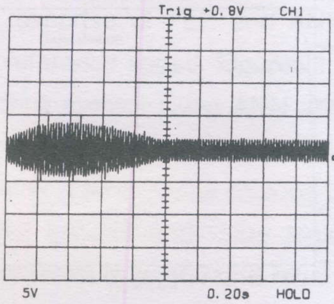


Fig.8 (b) Experimental current of IPMSM drive at a reference speed of 188. 5 rad/s using FLC. 1 division=2 Amp

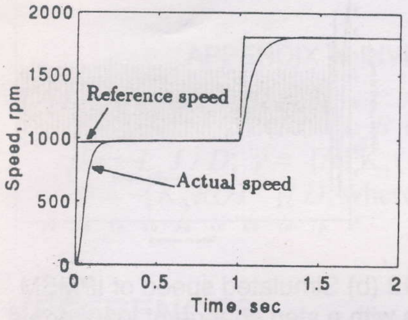


Fig. 9 (a) Simulated current of IPMSM drive with a step change of load torque using ANN.

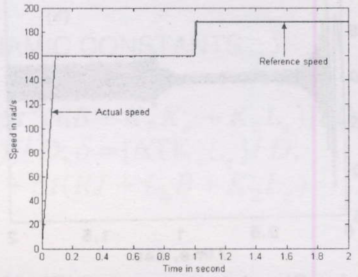


Fig.9 (b) Simulated current of IPMSM drive with a step change of load torque using FLC.

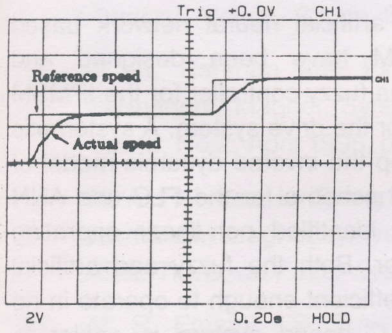


Fig.10 (a) Experimental speed of IPMSM drive with a step change of reference speed using ANN. 1 division=65.5 rad/s

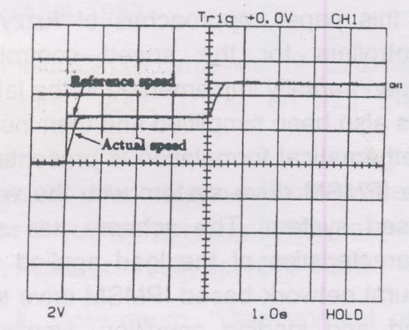


Fig.10 (b) Experimental speed of IPMSM drive with a step change of reference speed using FLC. 1 division=65.5 rad/s

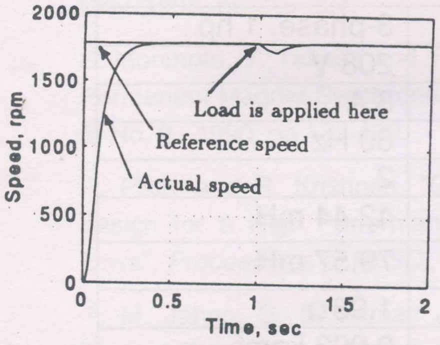


Fig.11 (a) Simulated speed of IPMSM drive with a step change of reference speed using ANN.

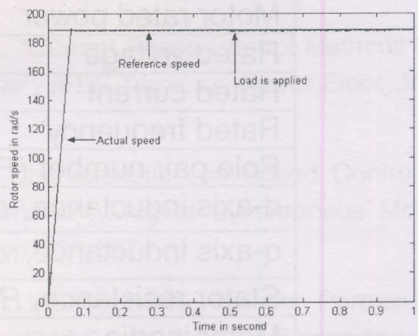


Fig. 11(b) Simulated speed of IPMSM drive with a step change of reference speed using FLC.

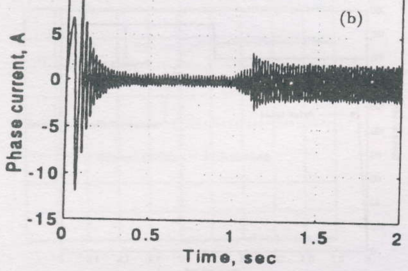


Fig.12 (a) Simulated speed of IPMSM drive with a step change of load torque using ANN.

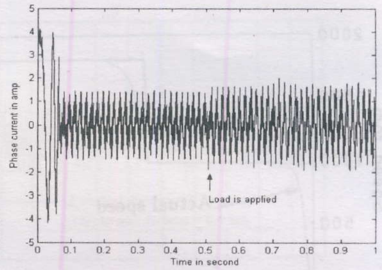


Fig.12 (b) Simulated speed of IPMSM drive with a step change of load torque using FLC.

7. CONCLUSIONS

In this paper, approaches of fuzzy logic and artificial neural network based controllers for the speed control of IPMSM have been designed and experimentally implemented in the laboratory. The fuzzy controller for the IPMSM has also been simplified and then incorporated for the drive system. A systematic mathematical formulation is presented to develop the inverse dynamic model of the IPMSM drive system with the vector control scheme for the FLC and ANN based system. The scheme has successfully identified non-linear operating characteristics of the load applied to the motor. Both the fuzzy and artificial neural network based IPMSM drive system are efficient enough to operate in no load and loading condition. However, the FLC based system is easier to implement in the laboratory and it requires lower settling time than those of ANN based drive system. From the obtained results, it is obvious that the FLC based IPMSM drive has been found superior to the ANN based system.

Table-4: Machine parameters

Motor rated power	3-phase, 1 hp
Rated voltage	208 V
Rated current	3 A
Rated frequency	60 Hz
Pole pair number (*P*)	2
d-axis inductance, L_d	42.44 mH
q-axis inductance, L_q	79.57 mH
Stator resistance, *R*	1.93 Ω
Motor inertia, J_m	0.003 kgm^2
Friction coefficient, B_m	0.001 Nm/rad/sec
Magnetic flux constant, ψ_f	0.311 volts/rad/sec

APPENDIX A: INVERSE DYNAMIC CONSTANTS

$$\alpha = [2L_q J + \Delta T(RJ + L_q B + K_2 L_q) - \Delta t^2 (RB + K_T K_b + K_2 L_q)] / D;$$
$$\beta = -L_q J / D;\ \gamma = -[\Delta \mathrm{TK}_1 (L_q + R\Delta T)] / D; \delta = [\Delta \mathrm{TK}_1 L_q] / D;$$
$$\vartheta = -[K_3 ROT^2] / D; \text{where } \mathrm{D} = \mathrm{L_q}\, j + \Delta t(RJ + L_q B + K_2 L_q)$$

REFERENCES

[1] R. Krishnan, "Selection criteria for servo motor drives", *IEEE Trans. Ind. Applicat.*, Vol. IA- 23, pp. 270-275, Mar/April, 1987

[2] M. A. Rahman and Ping Zhou, "Field Circuit analysis of Brushless Permanent Magnet Synchronous Motors", IEEE Trans. on Ind. Elect. Vol.43, No.2, April 1996, pp. 256-267.

[3] M. A. Rahman, L. Zhong and K.W. Lim, "A direct torque-controlled interior permanent magnet synchronous motor drive incorporating field weakning", *IEEE Trans. on Fuzzy System,*Vol. 34, Nov/Dec. 1998, pp.1246-1253

[4] M. R. Emami, I.B. Turksen and A. A. Goldenburg, "Development of A Systematic Methodology of Fuzzy Logic Modeling", *IEEE Trans. on Ind. Applicat.,*Vol. 6, No. 3, August. 1998, pp.346-361.

[5] S. Y. Yi and M. J. Chung, "Robustness of Fuzzy Logic Control for an Uncertain Dynamic System", *IEEE Trans. on Fuzzy System.,*Vol. 6, No.2, May. 1998, pp.216-224.

[6] S. Morimoto, Y. Takeda and T. Hiras, "Current Phase Control Methods for Permanent Magnet Synchronous Motor", *IEEE Trans. on Power Elect..,*Vol. 5, No. 2, 1990, pp.133-139.

[7] P. Pillay and R. Krishnan, "Control Characteristic and Speed Controller design for a High Performance Permanent Magnet Synchronous Motor Drive", *Proceedings of PESC,,* 1987, pp. 598-607.

[8] T. M. Jahns, G. B. Kliman and T. W. Neumann, "Interior Permanent Magnet Synchronous Motors for Adjustible Speed Drives", *Proceedings of IEEE/IAS Meeting* , 1985, pp.814-823.

[9] B. K. Bose, "A high performance inverter fed drive system of an interior permanent magnet synchronous machine", *IEEE Trans. on Ind. Applicat.,*Vol. IA-24, No. 6, Sep/Oct. 1988, pp.987-997.

[10] N. N. Uddin and M. A. Rahman, "Fuzzy logic based speed control of an IPM synchronous motor drive", J. Advanced Comput. Intell., vol.4, no.3, pp. 212-219, 2000.

[11] M. A. Hoque, M. R. Rahman and M. A. Rahman, "Artificial neural network based permanent magnet dc motor drives", *Conference Record, IEEE/IAS Annual Meeting* , Vol. 2., 1995, pp.1775-1780.

[12] J. Theocharis and V. Petridis, "Neural network observer for induction motor control", *IEEE control system Magazine* , 1994, pp.26-37.

[13] B. Burton, F. Kamran, R.G. Harley, G.T. Habetter, M. Broike, and R. Poddar, "Identification and control of induction motor stator currents using fast on-line random training of a neural network" *Conference Record, IEEE/IAS Annual Meeting* , Vol. 2., 1995, pp.1781-1787.

[14] M. A. Rahman and M. A. Hoque "On-line adaptive artificial neural network based vector control of permanent magnet synchronous motors" *IEEE Trans. on Energy Convrsion,* Vol. 13, No.4,. Dec. 1998, pp.311-318.

[15] H. Demuth and M. Beale, "Neural Network control tool-box user's guide", The Math Works Inc., 1993.

[16] L. A. Zadeh, "Fuzzy sets," *Inform. Control*, vol. 8, pp. 338–353, 1965.