Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 5 (7): 1286-1291, 2005
ISSN 1812-5654

A PI Type Fuzzy-neural Network Controller for Induction Motor Drives

[1]Beşir Dandil, [1]Muammer Gökbulut and [2]Fikret Ata
[1]F.Ü. Tek. Eğt. Fak. Elektronik ve Bilgisayar Eğt. Böl.
[2]F.Ü. Mühendislik Fak. Elektrik-Elektronik Müh. Böl. Elaziğ, Turkey

Abstract: For high performance electrical drives, a desirable control performance must be provided even when the parameters and load of the motor are varying during the motion. This study proposed a Proportional Integral (PI) type Fuzzy-Neural Network (FNN) controller for a vector controlled induction motor drives to deal with these issues. The fuzzy-neural controller based on Sugeno fuzzy model was adopted for this study and FNN inputs were selected as the speed error and the error integral to eliminate steady state errors. Experimental results showed the speed control performance of the proposed control system was presented for various operating conditions of the motor.

Key words: Fuzzy-neural network controller, induction motor drives, vector control

INTRODUCTION

High accuracy is not usually imperative for most electrical drives, however, in high performance drive applications, a desirable control performance in both transient and steady states must be provided even when the parameters and load of the motor are varying during the motion. Controllers with fixed parameters can not provide these requirements unless unrealistically high gains are used. Thus, the conventional constant gain controllers used in the high performance variable speed drives become poor when the uncertainties of the plant exist, such as load disturbance, mechanical parameter variations and unmodelled dynamics in practical applications[1,2]. Therefore, control strategy of high performance electrical drives must be adaptive and robust. As a result, interest in developing adaptive control methods for electrical drives has increased considerably with in the last two decades and several adaptive control methods based on linear model have been developed for induction motor drives[3,4].

In the past decade, fuzzy logic and neural network control techniques have been applied to electrical drives to deal with nonlinearities and uncertainties of the control system. Fuzzy control has the ability of implementing expert human knowledge and experience expressed in the form of linguistic rules. It is easy to understand the structure of the fuzzy controller and to modify the control laws. Hence, fuzzy logic control introduces a good tool to deal with the complicated, nonlinear and ill-defined systems which cannot be described by precise mathematical models. However, fuzzy controllers have difficulties in determining suitable fuzzy control laws and tuning the parameter of the membership functions for system changes[5-7]. The major advantageous features of neural network are their learning and generalization capability and fault tolerance. It can adapt itself to changing control environment using the system input and output and it does not require complicated control theories and exact knowledge of the system. However, neural network has some problems in training: the sensitivity of the controlled system which is difficult to obtain for unknown and nonlinear systems is required and the local minimum of the performance index can be trapped. Besides, it is difficult for the user to decide the structure of the neural network for the desired control[7].

Fuzzy-Neural Network (FNN) approach incorporates the fuzzy logic controller into the neural network structure. Neural network provides connectionist structure and learning abilities to the fuzzy logic controller. In recent years, FNN control is applied to induction motors[8-10] and used to update the control gain of the sliding mode position controller for an induction motor drive[11]. Fuzzy-neural network controller is augmented with an IP controller[12], PD controller[13] and an adaptive controller[14]. In this study, a PI type FNN controller based on Sugeno fuzzy model is proposed for induction motor drives. The FNN controller uses the speed error and error integral as inputs and gives the torque current command as output. The backpropagation algorithm is used to train the FNN online in the direct adaptive control scheme. Speed control performance of the proposed control system is evaluated under the parameter and load variations of the motor using the experimental setup including the DSPACE-1104 signal processor control card.

Corresponding Author: Dr. Beşir Dandil, F.Ü. Tef. Elektrik Egitimi Balumu, Elaziğ, Turkey
Tel: +90 42 42470000/6654

FUZZY-NEURAL NETWORK CONTROL OF INDUCTION MOTORS

The mathematical model of a three phase Y-connected squirrel cage induction motor is given in the synchronously rotating d-q reference frame by the following set of equations:

$$V = AI \tag{1}$$

$$V = [v_{ds} \quad v_{qs} \quad 0 \quad 0]^T \tag{2}$$

$$I = [i_{ds} \quad I_{qs} \quad I_{dr} \quad I_{qr}]^T \tag{3}$$

$$\mathbf{A} = \begin{bmatrix} R_s+pL_s & -\omega_e L_s & pL_m & -\omega_e L_m \\ \omega_e L_s & R_s+pL_s & \omega_e L_m & pL_m \\ pL_m & -\omega_{sl} L_m & R_r+pL_r & -\omega_{sl} L_r \\ \omega_{sl} L_m & pL_m & \omega_{sl} L_r & R_r+pL_r \end{bmatrix} \tag{4}$$

Where, ω_e and ω_r are synchronous speed and rotor speed, respectively and slip frequency is $\omega_{sl} = \omega_e - \omega_r$. The rotor flux orientation implies that $\lambda_{dr} = \lambda_r$ and $\lambda_{qr} = 0$. Then, two important relations can be derived as following. The required slip frequency can be calculated as a linear function of the stator q axis (torque) current and an inverse function of the d axis (flux) current:

$$\omega_{sl} = \frac{L_m}{\tau_r}\frac{i_{qs}}{\lambda_{dr}} = \frac{1}{\tau_r}\frac{i_{qs}}{i_{ds}} \tag{5}$$

The electromagnetic torque is a linear function of the stator q axis current and the rotor flux:

$$T_e = \frac{3}{2}\frac{P}{2}\frac{L_m}{L_r}\lambda_{dr} i_{qs} = K_T . i_{qs} \tag{6}$$

where, K_T is the torque constant. Block diagram of the induction motor drive including the proposed FNN controller is shown in Fig. 1, which consists of a induction motor loaded with a DC generator, current controlled PWM voltage source inverter, vector control mechanism and a speed control loop. The control algorithm, current control and PWM generation is realized in a PC including DSPACE-1104 signal processor control card.

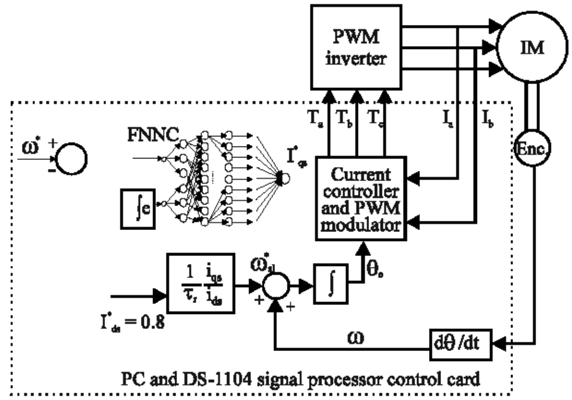

Fig. 1: Block diagram of the proposed control system

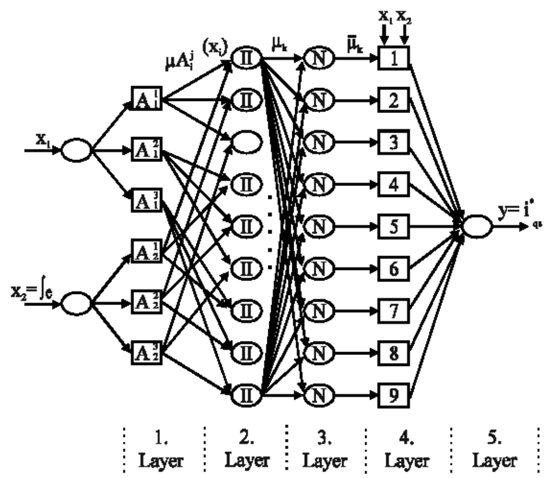

Fig. 2: Structure of fuzzy-neural network controller

ARCHITECTURE OF FNN CONTROLLER

Sugeno type FNN controller as shown in Fig. 2 is adopted for this study. For a first order Sugeno FNN, a common rule set with two fuzzy if-then rules is the following[5,7]:

$$\begin{aligned} &R^1\text{:IF } x_1 \text{ is } A_1^1 \text{ and } x_2 \text{ is } A_2^1, \text{ then } y = f_1 = a_0^1 + a_1^1 x_1 + a_2^1 x_2 \\ &R^2\text{:IF } x_1 \text{ is } A_1^2 \text{ and } x_2 \text{ is } A_2^2, \text{ then } y = f_2 = a_0^2 + a_1^2 x_1 + a_2^2 x_2 \end{aligned} \tag{7}$$

Where, x_i is the input variable, y is the output variable, A_i^j are linguistic variables of membership functions $\mu_{A_i^j}(x_i)$ and $a_i^j \in R$ are parameters of the linear output function $f_i(x_1, x_2, \ldots, x_n)$, which are called as consequent parameters.

FNN inputs was selected as the speed error $x_1 = e(t)$ and the integral of the error $x_2 = \int e(t)$, where $e(t) = \omega^*(t) - \omega(t)$ and ω^* is the reference speed and ω is actual rotor speed. The input layer transmits input signals to the first layer. Every node in the first layer acts as a membership function $\mu_{A_i^j}(x_i)$ and its output specifies the degree to which to given x_i satisfies the quantifier A_i^j .

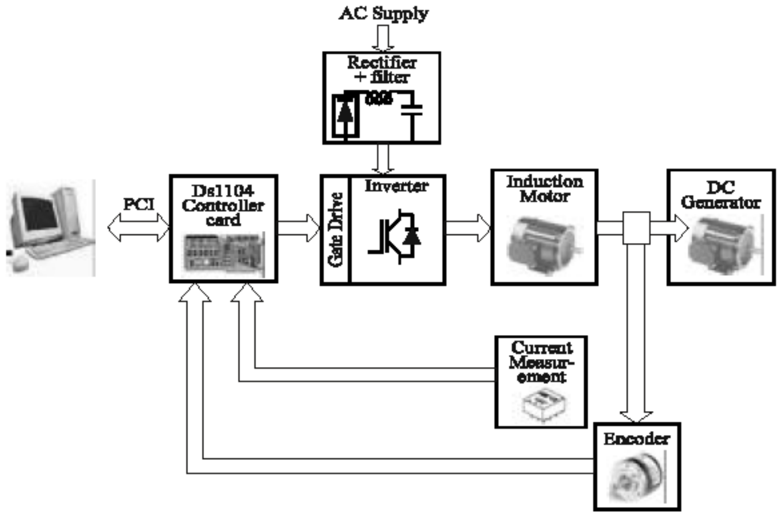

Fig. 3: Block diagram of the experimental rig

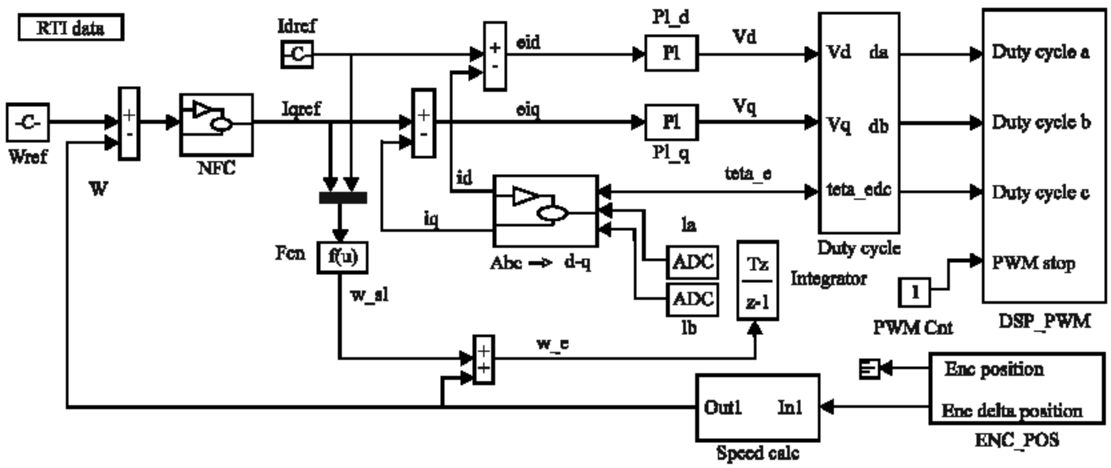

Fig. 4: Experimental implementation of the control system using MATLAB/Simulink

$$\mu_{A_i^j}(x_i)=\exp\left\{-\left[\left(x_i-m_i^j\right)/\sigma_i^j\right]^2\right\}^{b_i^j} \quad (8)$$

where, $\{m_i^j,\sigma_i^j,b_i^j\}$

are parameters of the membership function $\mu_{A_i^j}(x_i)$, which are called as premise parameters. Every node in the second layer was labeled Π and performs fuzzy and operation. Every node in this layer was a fixed node, which operates the incoming signal from every set of the membership function nodes for their for their corresponding input. Each node output represents the firing strength of a rule.

$$\mu_k=\min(\mu_{A1}^{\,j}(x_1),\ \mu_{A2}^{\,j}(x_2)) \quad (9)$$

Every node in the third layer was labeled N and it calculates the normalized firing strength of a rule. That was, kth node calculates the ratio of the kth rule's firing strength to the sum of all rule's firing strength;

$$\bar{\mu}_k=\frac{\mu_k}{\sum\mu_k} \quad (10)$$

Every node k in the fourth layer calculates the weighted consequent value $\bar{\mu}_k f_k$, where $\bar{\mu}_k$ is the output of layer 4 and f function is,

$$f_k=a_0^k+x_1a_1^k+x_2a_2^k \quad (11)$$

where, $\{a_0^k,a_1^k,a_2^k\}$ is parameter set which are referred to consequent parameters. The only node in the fifth layer is labeled $\sum$ and it sums all incoming signals to obtain the final inferred result for the whole system.

$$y=\frac{\sum\mu_k f_k}{\sum\mu_k}=\sum\bar{\mu}_k f_k \quad (12)$$

Backpropagation algorithm is used to update the premise and consequent parameters of the FNN. Premise and consequent parameters of the FNN are modified as

$$\frac{\partial E}{\partial a_0^k} = \delta^1 \frac{\mu_k}{\sum \mu_k} \qquad (13)$$

$$\frac{\partial E}{\partial m_i^j} = \delta_1 f_k \frac{\sum \mu_k - \mu_k}{(\sum \mu_k)^2} \prod \mu_{A_i^j}(x_i) \left[2b \frac{(x_i - m_i^j)}{\sigma^2} \right] \mu_{A_i^j}(x_i) \qquad (14)$$

where, $\delta^1 = \partial E / \partial Y$ is the local gradient calculated from the system dynamics.

EXPERIMENTAL RIG

The FNN system proposed in this study was implemented using the dSPACE-DS1104 signal processor control card. DS1104 produces PWM signals for the inverter using the stator currents and rotor position measured from the current sensors and encoder unit, respectively.

DS-1104 control card includes master processor of PowerPC 603e/250MHz and slave-processor of Texas Instruments TMS320F240 (Fig. 3). The control algorithm, current control and PWM modulation is realized in a PC with dSPACE-1104 control card. dSPACE-DS1104 control card allows user to construct the system in MATLAB/Simulink and then to convert the model files to real-time codes using the Real-Time Workshop of the MATLAB/Simulink and Real-Time Interface (RTI) of the dSPACE-DS1104 control card. The RTI software comprises of four sub-libraries, (dSPACE RTI1104), including some sub-blocks which provide the connection between Simulink and physical equipment such as; digital-analog converter, analog-digital converter, incremental encoder interface and various pulse with modulation units. These blocks are added to Simulink libraries by RTI. Hence, experimental implementation of the control system is realized using Matlab/Simulink diagram as shown in Fig. 4.

Real time values of the physical systems' variables can be assigned to the user defined variables using the dSAPCE-Control Desk Developer (CDD) software. Thus the graphical user interface can be designed by the user, to observe the real time values of the variables or to change the input variables such as reference speed.

EXPERIMENTAL RESULTS

Some experimental results were provided to demonstrate the effectiveness of the proposed fuzzy-

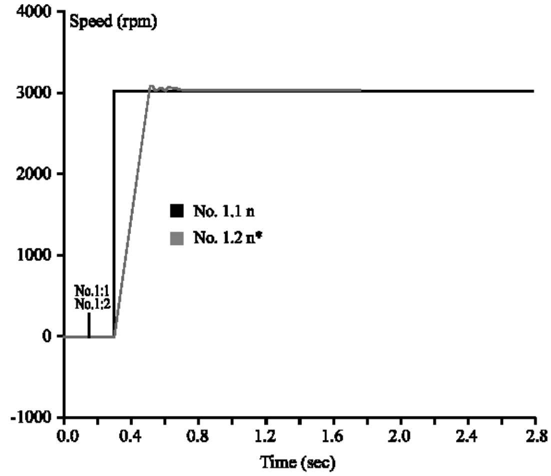

Fig. 5: Step response of the motor for no load condition

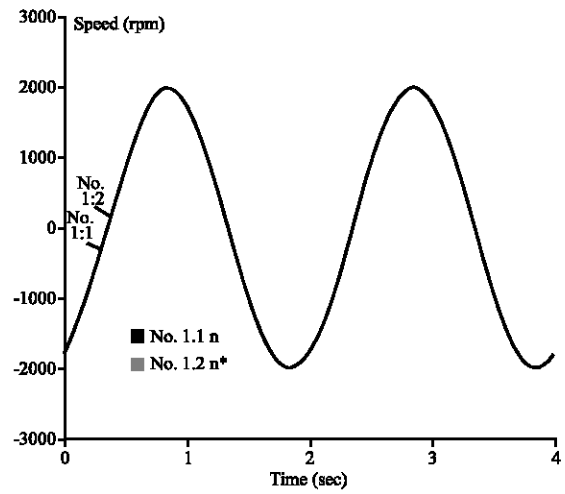

Fig. 6: Sinusoidal speed response of the motor for no load condition

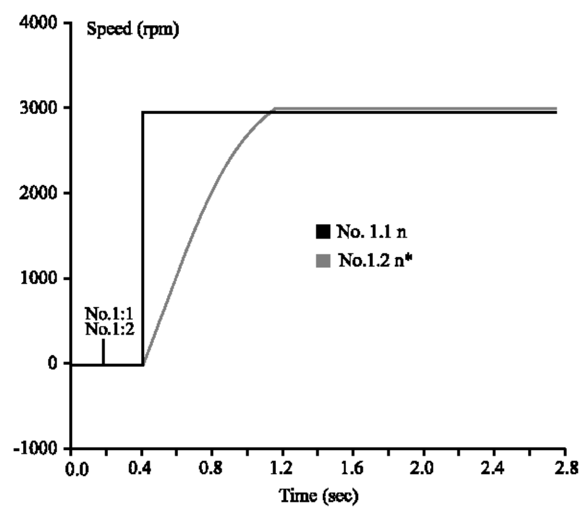

Fig. 7: Step response for the increased inertia

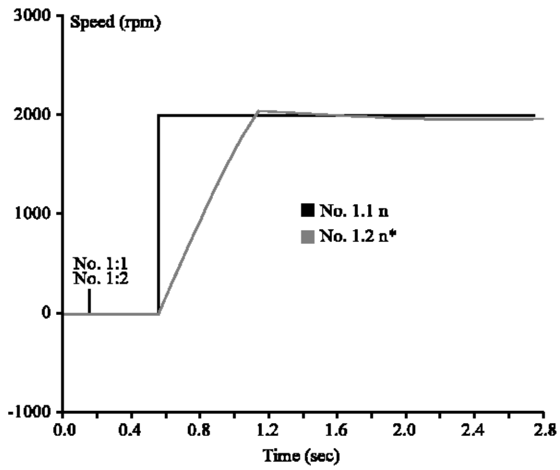

Fig. 8: Step response of the motor for 0.9 pu load condition

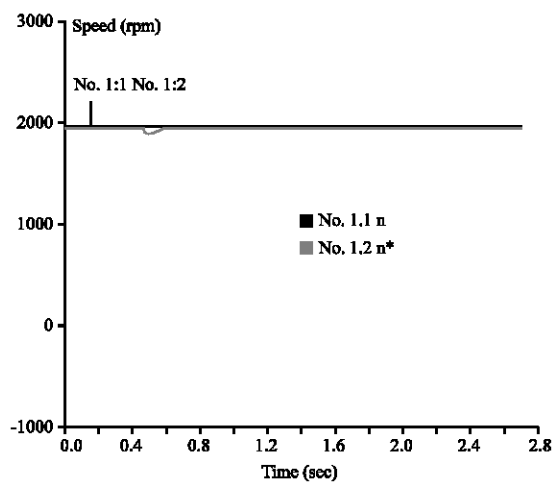

Fig. 9: Step response of the motor for 0.9 pu load disturbance

neural controller. Sampling rate of current and speed control loop was 70 μs and 700 μs, respectively. FNN controller was trained online using the simulation model of the motor and then trained FNN controller was used for experiments. Tracking performances of the FNN controller were tested for various load conditions and mechanical parameter variations. First, tracking response for no load condition is given in Fig. 5 for step reference and in Fig. 6 for sinusoidal reference. In the second experiment, inertia of the motor was increased by a coupled disc about four times of the nominal value and the speed tracking response is shown in Fig. 7 for step reference. As the mechanical time constant of the drive was increased, rise time was increased compared to Fig. 5. In the third experiment, the controller was tested under with the speed dependent load produced by the DC generator. The maximum value of the load was 90% of the nominal value. The speed tracking response is shown in Fig. 8. Finally, Fig. 9 shows the performances of the controllers when 90% load disturbances was applied. As seen in the Fig.5-9, excellent tracking performance was obtained with no steady state error and no overshoot and control performance of the drive is acceptable for load disturbance.

CONCLUSIONS

In this study, FFN approach was applied to induction motor drive. PI-type FNN based on Sugeno fuzzy model was adopted for this application in direct adaptive control scheme. Speed error and error integral were selected as inputs to the FNN, to eliminate the steady state error. FNN was trained online using the simulation model of the motor and then trained FNN was used in experiments. Experimental results showed the effectiveness of the FNN were presented for various load conditions.

Motor parameters: P=1.1kW, V=220V, P=2, f=50Hz, T=3.72N.m R_S=8.5Ω, R_r=4.59Ω, L_S=0.5999H, L_r=0.5999H, L_M=0.5787H, J=0.0019, B=0.000263.

REFERENCES

1. Leonard, W., 2001. Control of electrical drives. Spring Verlag, Berlin, pp: 252-260.
2. Bose, B.K., 2002. Modern power electronics and AC drives. Prentice-Hall, Upper Saddle River, NJ., 56: 413-419.
3. Cerruto, E., A. Consoli, A. Raciti and A. Testa, 1995. A robust adaptive controller for PM motor drives in robotic application. IEEE Trans. Power Electron., 10: 62-71.
4. Liu, T.H. and C.P. Cheng, 1993. Controller design for a sensorless permanent magnet synchronous drive system. IEE Proc. Pt B, 140: 369-378.
5. Lin, C.T. and C.S. Lee, 1996. Neural Fuzzy Systems. Prentice Hall PTR, New Jersey, pp:551-559.
6. Jang, J.S.R., 1993. ANFIS: Adaptive-network-based fuzzy inference systems. IEEE Transactions on Systems, Man and Cybernetics, 23: 665-685.
7. Jang, J.S.R., C.T. Sun and E. Mizutani, 1997. Neuro-Fuzzy and Soft Computing, Simonand Schuster. Prentice Hall Inc., Upper Saddle River, pp: 333-342.
8. Vas, P. and A.F. Stronach, 1996. Adaptive fuzzy-neural DSP control of high-performance drives. Power Electronics and Variable Speed Drives Conf., 23-24 September 1996, pp: 424-429.

9. Aware, M.V., A.G. Kothari and S.O. Choube, 2000. Application of Adaptive Neuro-Fuzzy Controller (ANFIS) for voltage source inverter fed induction motor drive. Power Electronics and Motor Control Conf., 2: 935-939.
10. Dandil, B., 2004. Robust speed control of induction motors using neuro-fuzzy controllers. Ph.D Thesis, Firat University, Electrical and Electronics Engineering.
11. Wai, R.J. and F.J. Lin, 1999. Fuzzy-neural network sliding mode position controller for induction motor drive. IEE Proc. Electrical Power Applied, 146: 297-308.
12. Lin, F.J., R.J. Wai and H.P. Chen, 1998. A PM synchronous servo motor drive with an on-line trained fuzzy-neural network controller. IEEE Trans. on Energy Conversion, 13: 319-325.
13. Er, M.J. and Y. Gao, 2003. Robust adaptive control of robot manipulators using generalized fuzzy neural networks. IEEE Transactions on Industrial Electronics, 50: 620-628.
14. Lin, F.J. and R.J. Wai, 2002. Adaptive fuzzy-neural-network control for induction spindle motor drive. IEEE Trans. on Energy Conversion, 17: 507-513.