DYNAMIC SPEED CONTROL OF THREE PHASE INDUCTION MOTOR USING ARTIFICIAL INTELLIGENCE BASED ON ANFIS

Nandini.P[1] | Vaikundaselvan.B[2] | Venupriya.S.R[3]

[1](P.G Student, Kathir College Of Engineering, Coimbatore-641062, Tamil Nadu, India, nandinimayflower@gmail.com)
[2](Professor, Dept. of EEE, Kathir College Of Engineering, Coimbatore-641062, Tamil Nadu,India,eeehod.kce@gmail.com)
[3](Assistant Professor, Kathir College Of Engineering, Coimbatore-641062, Tamil Nadu, India, venupriyaeee@gmail.com)

Abstract—*Induction motors are commonly used in industries due to low maintenance and robustness. By controlling the speed of Induction motor maximum efficiency and torque can be obtained. Using artificial intelligence particularly Fuzzy and Neural Networks, Induction motor performance can be improved. This paper presents dynamic speed control of induction motor drive using ANFIS. The integrated solution allows the user to compare the Neural Network and ANFIS technique. By using ANFIS the applied voltage frequency is controlled and thus the speed of the Induction motor is controlled to the required value. Rise time of the motor is decreased and pick – up speed is increased. By this the performance of the Induction motor is increased. The dynamic modelling and simulation of induction motor has been done using MATLAB/SIMULINK and the Induction motor drive performance has been analyzed for Artificial Intelligence controller.*

Keywords—Neuro Network(NNW), ANFIS Controller, Induction Motor, Fuzzy Logic

1. INTRODUCTION

Induction Motors are often called the “Workhorse of the Industry”. This is because it is one of the most widely used motors in the world. It is widely used in industries and transportation, also in household appliances, and laboratories. Three phase Induction Motor have extensive applications in electrical machines. About half of the electrical energy generated in a developed country is ultimately consumed by electric motors, of which over 85% are induction motors. For the high power drives in industries, the lighter, less expensive, reliable simple, more robust and commutator less induction motors .In about the same period, there were also advances in control methods and Artificial Intelligence (AI) techniques. Artificial Intelligent techniques make use of fuzzy logic, neural networks, genetic algorithm and expert system. It is realized that the performance of induction motor drives can be improved by adopting artificial-intelligence -based methods. The Artificial Intelligence (AI) techniques, such as Genetic Algorithm (GA), Fuzzy Logic (FL), Artificial Neural Network (ANN), and Expert System (ES) have recently been applied broadly in control of induction motor drives. Among all the branches of AI, the ANFIS tend to have greater impact on motor drives area that is evident with the aid of the guides in the literature. This paper tends to show ANFIS has better performance over fuzzy controller.

2. ADAPTIVE NEURO – FUZZY INFERENCE SYSTEM

This section describes a class of Neuro-Fuzzy along with the architectures and learning procedures of adaptive networks. The underlying network structure is a superset of all kinds of neural network paradigms with supervised learning capability. Neuro-fuzzy systems, is the combination of ANN with fuzzy systems, usually have the benefit of allowing an easy translation of the final system into a set of if-then rules, and the fuzzy system can be viewed as a neural network structure with knowledge distributed throughout connection strengths. Research and applications on neuro-fuzzy inference strategy made clear that neural and fuzzy hybrid systems are beneficial in fields such as the applicability of existing algorithms for artificial neural networks (ANNs)[7], and direct model to know-how knowledge is expressed as as a set of fuzzy linguistic rules. An adaptive network, as its name implies, is a network structure consisting of nodes and directional links, overall input-output behavior is determined by the values of a collection of modifiable parameters through which the nodes are connected. The adaptive system uses a hybrid learning algorithm to identify parameters of Sugeno-type fuzzy inference systems.

It applies a combination of the least-squares method and the back-propagation gradient descent method for training FIS membership function parameters to emulate a given training data set .The network learns in two main phases. In the forward phase of the learning algorithm, consequent parameters identify the least squares estimate. In the backward phase, the error signals, which are the derivatives of the squared error with respect to each node output, propagate backward from the output layer to the input layer. In this backward pass, the premise parameters are updated by the gradient descent algorithm. Learning or training phase of the neural network is a process to determine parameter values to sufficiently fit the training data. ANFIS training can use alternative algorithms to reduce the error of the training. A combination of the gradient descent algorithm and a least squares algorithm is used for an effective search for the optimal parameters. The main advantage of such a hybrid approach is that it converges much faster, since it reduces the search space dimensions of the back propagation method used in neural networks . ANFIS are the fuzzy Sugeno model put in framework of the adaptive system which serves in model building and validation of developed model to facilitate training and adaptation.

3. ARCHITECTURE OF ANFIS

An adaptive network is a multilayer feed-forward network composed of nodes connected by directed links, in which each node performs a particular function on its incoming signals to generate a single node output. Each link in an adaptive network specifies the direction of signal flow from one node to another; no weights is associated with the link. More specifically, the configuration of an adaptive network performs a static node function on its incoming signals to generate a single node output and each node function is a parameterized function with modifiable parameters; by changing these parameters, the node functions as well as the overall behavior of the adaptive network, are changed.

Entire system architecture consists of five layers, namely fuzzy layer, product layer, normalized layer, de-fuzzy layer and total output layer. With input/output data for given set of parameters, the ANFIS method models a fuzzy inference system (FIS) whose membership function parameters are tuned (adjusted) using either a back-propagation algorithm alone, or in permutation with a least squares type of method. The main objective of the ANFIS is to determine the optimum values of the equivalent fuzzy inference system parameters by applying a learning algorithm. The parameter optimization is done in such a way during the training session that the error between the target and the actual output is minimized.

A hybrid algorithm is used for optimization, which is the combination of least square estimate and gradient descent method. The parameters to be optimized in ANFIS are the premise parameters. These parameters define the shape of the membership functions . In order to reduce the error measure, any of several optimization routines can be applied after constituting MFs. The parameter set of an adaptive network allows fuzzy systems to learn from the data they are modeling. This paper assumes that adaptive system under consideration has two inputs V1 and V2 and one output f. Let us scrutinize a first order Takagi, Sugeno and Kang (TSK) fuzzy inference system containing two rules:

Rule 1: If (v is V1) and (d is D1) then $f1 = p1v + q1d + r1$

Rule 2: If (v is V2) and (d is D2) then $f2 = p2v + q2d + r2$

Where p1, p2, q1, q2, r1 and r2 are linear parameters and V1, V2, D1 and D2 are non linear parameters, in which V1 and D1 are the membership functions of ANFIS (antecedent). p1, q1, r1 are the consequent parameters . To reflect adaptive capabilities, we use both circle and square. A circle indicates fixed node whereas square indicates adaptive node i.e. the parameter can be changed during adapting or training. ANFIS is created from integration of fuzzy logic and neural network.

While designing of ANFIS model, it is extremely important that the number of training epochs, the number of membership functions and the number of fuzzy rules should be tuned accurately. Mapping of those parameters is highly crucial for the system because it may lead system to over fit the data or will not be able to fit the data.

This adjusting can be obtained by using a hybrid algorithm combining the least-squares method and the gradient descent method with a mean square error method. The lesser difference between ANFIS output and the desired objective means a better (more accurate) ANFIS system.

So we tend to reduce the training error in the training process . The integration between fuzzy logic and neural network namely fuzzy neural network (FNN) has been expected and developed; generally the arrangement of fuzzy logic and the neural network is called as ANFIS. Neural system has many inputs and also has multiple outputs, but the fuzzy logic has copious inputs and single output, so the combination of this two is known as ANFIS.

3.1 Layers of ANFIS

For simplicity, the fuzzy inference system is under consideration of two inputs v, d and one output f. A brief summary of four layers of the ANFIS algorithm is shown below.

3.1.1 Layer 1

Each input node i in this layer is an adaptive node which produce membership grade of linguistic label.

It is a fuzzy layer, in which v and d are input of system. O1,i is the output of the ith node of layer l. Each adaptive node is a square node with square function represented using Equation (3.4),

$$O1,i = \mu v,i(v) \text{ for } i = 1, 2 \quad (3.3)$$

$$O1,j = \mu d,j(v) \text{ for } j = 1, 2 \quad (3.4)$$

Where O1,i and O1,j denote output function and μv,i and μd,j denote membership function. For example if we choose triangular membership function, μv,i(v) is given by:

$$\mu vi(v) = \max \mathbf{min}\,(\boldsymbol{v{-}aibi{-}ai}, \boldsymbol{ci{-}vci{-}bi}), \mathbf{0} \quad (3.5)$$

Where {ai,bi,ci} are the parameter of triangular membership function. In other example, if we choose μv,i(v) to be bell shaped is given by:

$$\mu\, \mathbf{v} = \mathbf{11}{+}\{(\,\boldsymbol{v{-}ciai}\,)\mathbf{2}\}\boldsymbol{bi} \quad (3.6)$$

Where {ai,bi,ci} are the parameter set that changes shapes of M.F accordingly. Value of ai and ci that can be adjusted to vary the center and width of membership function and then bi is used to control slopes at crossover points of next membership function. Parameters in this layer are referred to as “premise parameter”.

3.1.2 Layer 2

This layer checks weights of each membership function, it receives input values vi from first layer and acts as a membership function to represent fuzzy sets of respective input variables. Every node in this layer is fixed node labeled with M and output is calculated via product of all incoming signals. The output in this layer can be represented using Equation 3.7:

$$O2,i = wi = \mu v,i(v).\ \mu Dj(d)\ ,\ i = 1, 2 \quad (3.7)$$

Which are the firing strengths of the rules. In general, any T-norm operator that performs fuzzy AND can be used as a node function in this layer.

3.1.3 Layer 3

Every node in this layer is fixed marked with circle labeled with N, indicating normalization to the firing strength from previous layer. This layer performs pre-condition matching of fuzzy rules, i.e. they compute activation level of each rule, the number of layers being equal to number of fuzzy rules. The ith node in this layer calculate ratio of ith rule‟s strength to the sum of all rules firing strength. The output of this layer can expressed as wi using Equation 3.8:

$$O3,i = \boldsymbol{wi} = \mathbf{wi}\boldsymbol{w}\mathbf{1}{+}\boldsymbol{w}\mathbf{2}\ ,\ i = 1, 2 \quad (3.8)$$

For convenience, outputs of this layer will be called as normalized firing strengths.

3.1.4 Layer 4

This layer provides output values y, resulting from the inference of rules. The resultant output is simply a product of normalized firing rule strength and first order polynomial. Weighted output of rule represented by node function as:

$$O4,i{=}\boldsymbol{wi}\text{fi}{=}\boldsymbol{wi}(\text{piv+qid + ri})\ ,\ i = 1, 2 \quad (3.9)$$

Where O4,i represents layer 4 output. In this layer, pi, qi and ri are linear parameter or consequent parameter.

3.1.5 Layer 5

This layer is called output layer which sums up all the inputs coming from layer 4 and transforms fuzzy classification results into crisp values This node computes summation of all incoming signals calculated using Equation 3.10,

$$O5,i{=}\boldsymbol{wifii}{=}\boldsymbol{wifiiw}\mathbf{1}{+}\boldsymbol{w}\mathbf{2}\ ,\ i = 1, 2 \quad (3.10)$$

Thus, it is observed that when the values of premise parameter are fixed, the overall output of the adaptive network can be expressed as linear combination of a consequent parameter. Constructed network has exactly the same function as a Sugeno fuzzy model. Overall output of a system (z) can be expressed as in Equation 3.11. It can be observed that ANFIS architecture consists of two adaptive layers, namely the first layer and the fourth layer. The three modifiable parameters {ai,bi,ci} are so-called premise parameter in first layer and in the fourth layer, there are also three modifiable parameters {pi,qi,ri} pertaining to the first order polynomial. These parameters are so-called consequent parameters.

$$z{=}\boldsymbol{w}\mathbf{1}\boldsymbol{w}\mathbf{1}{+}\boldsymbol{w}\mathbf{2}\boldsymbol{f}\mathbf{1}{+}\boldsymbol{w}\mathbf{2}\boldsymbol{w}\mathbf{1}{+}\boldsymbol{w}\mathbf{2}\boldsymbol{f}\mathbf{2}{+}\cdots{+}\boldsymbol{wnwn}{-}\mathbf{1}{+}\boldsymbol{wnfn} \quad (3.11)$$

4. ARTIFICIAL INTELLIGENCE CONTROLLER DESIGN

AI controller is a device which controls each and every operation in the system making decisions. As per the control system point, it is bringing stability to the system when there is a disturbance, thus safeguarding the equipment from further damages.

It may be hardware based controller or a software based controller or a combination of both. Fuzzy logic is one of the successful applications of fuzzy set in which the variables are linguistic rather than the numeric variables.

Linguistic variables, defined as variables whose values are sentences in a natural language (such as large or small) it may be represented by the fuzzy sets. Fuzzy set is an extension of a “crisp set where an element can only belong to a set (full membership) or not belong at all (no membership).

Fuzzy sets allow partial membership, which means that an element may partially belong to more than one set. A fuzzy set A of a universe of discourse X is represented by a collection of ordered pairs of generic element and its membership function

$\mu : X \quad \{0\ 1\},$

which associates a number

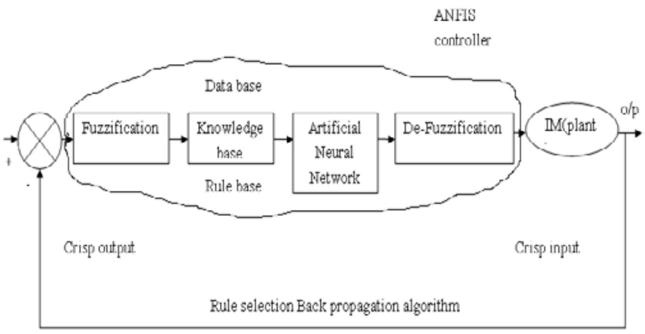


Figure 4.3 Block Diagram of the ANFIS Control Scheme

$A(x) : X \quad \{\ 0\ 1\}$, to each element x of X. A fuzzy logic controller is based on a set of control rules called as the fuzzy rules among the linguistic variables These rules are expressed in the form of conditional statements.

Our basic structure of the developed ANFIS coordination controller to control the speed of the Induction Motor consists of 4 important parts viz., fuzzification, knowledge base, neural network and the de-fuzzification blocks.

The inputs to the ANFIS controller, i.e., the error & the change in error is modeled by following equations

$$e(k)=\omega_{(ref-)}\ \omega_r \qquad (4.1)$$

$$\Delta e(k)=e(k)-e(k-1) \qquad (4.2)$$

Where, 〖 ω〗 _(ref)is the reference speed, ω_r is the actual rotor speed, e(k) is the error and delta e(k-1) is the change in error.

The fuzzification unit converts the crisp data into linguistic variables, which is given as inputs to the rule based block. The set of 25 rules are written on the basis of previous knowledge and experiences in the rule based block. The rule base block is connected to the neural network block. Back propagation algorithm is used to train the neural network to select the proper set of rule base.

The control signal developed due to the training and this training is a very important step in the selection of the proper rule base. Once the proper rules are selected & fired, the control signal required to obtain the optimal outputs is generated.

The output of the Neural Network unit is given as input to the de-fuzzification unit and the linguistic variables are converted back into the numeric form of data in the crisp form. In the fuzzification process, i.e., in the first stage, the crisp variables, the speed error & the change in error are converted into fuzzy variables or the linguistics variables. The fuzzification maps the two input variables to linguistic labels of the fuzzy sets.

The fuzzy coordinated controller uses the linguistic labels. Each fuzzy label has an associated membership function. The membership function of triangular type is used in our work .The inputs are fuzzified using the fuzzy sets & are given as input to ANFIS controller. The rule base for selection of proper rules using the back propagation algorithm.

The developed fuzzy rules 5x5=25 included in the ANFIS controller and is shown in the table 4.1. The control decisions are made based on the fuzzified variables in the above Table. The inference involves a set of rules for determining the output decisions. As there are 2 input variables & 5 fuzzified variables, the controller has a set of 25 rules for the ANFIS controller.

Out of these 25 rules, the proper rules are selected by the training of the neural network with the help of back propagation algorithm & these selected rules are fired.

Further, it has to be converted into numerical output, i.e., they have to be de-fuzzified. This process is called as defuzzification, which is the process of producing a quantifiable result in fuzzy logic.

The output of the defuzzification unit will generate the control commands which in turn is given as input called as the crisp input, to the plant through the inverter.

If here is any deviation in the controlled output which is crisp output, this is fed back & compared with the set value & the error signal is generated which is given as input to the ANFIS controller which in turn brings back the output to the normal value, thus maintaining stability in the system.

Table 4.1 Fuzzy rule set in ANFIS controller

		Δ_e				
		NB	**NS**	**ZZ**	**PS**	**PB**
e	PB	ZZ	NS	NS	NB	NB
	PS	PS	ZZ	NS	NS	NB
	ZZ	PS	PS	ZZ	NS	NS
	NS	PB	PS	PS	ZZ	NS
	NB	PB	PB	PS	PS	ZZ

5. SIMULATION & RESULTS

5.1 INTRODUCTION

The simulation details of induction motor drive using ANN and ANFIS are explained . The simulation induction motor control of both ANN and ANFIS was done using the software package MATLAB/SIMULINK.

5.2 MATLAB/SIMULINK BLOCK FOR THREE PHASE INDUCTION WITH ANN

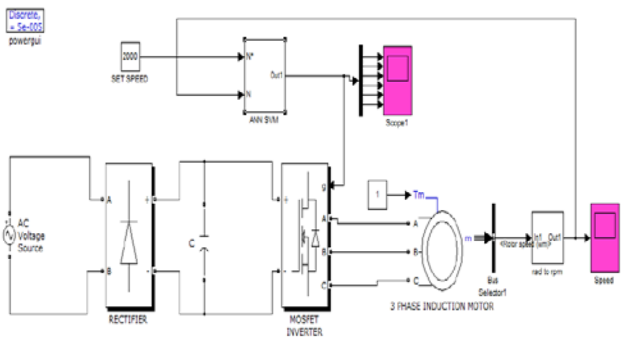


Figure 5.1 Three Phase Induction Motor Drive With ANN

The simulink block of three phase induction motor with ANN is Shown in the Figure.5.1.The AC source is first rectified to get the smooth waveform along with the filter to reduce distortion and noise in the acquired waveform then the rectified output is given to the VSI as an input and the output is given to the ANN SVM block for the selection of vectors. There frequency controlled voltage will be the output. Then by adjusting the input to the three phase induction motor by the out of inverter the speed can be varied accordingly.

5.3 SIMULATION MODEL OF ANN BLOCK

Simulation model of ANN block is shown in the Figure.5.2. In this block the reference speed and running speed error is compared and the numerical values of the voltage frequency is converted into the waveform by using oscillator then the oscillator output is given to the SVM block for the selection of vectors and thus speed of the induction motor is controlled by controlling the output of the inverter.

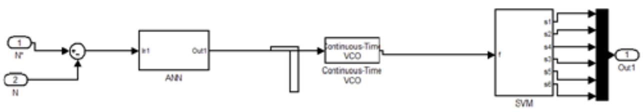


Figure 5.2 ANN Block for Controlling Three Phase Induction Motor

5.4 MATLAB/SIMULINK BLOCK FOR THREE PHASE INDUCTION WITH ANFIS

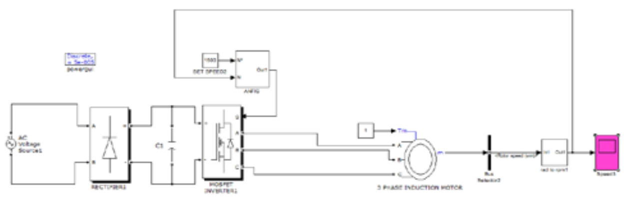


Figure 5.3 Three phase Induction motor drive with ANFIS

Simulink model for the control of various parameters of the induction motor can be developed in Matlab 11. By using the command window of Matlab it creates the .fis file and it will be supportive in the simulink for controlling the speed of Induction Motor with an important role of ANFIS controller proposed in this paper. From the various toolboxes available in the simulink library such as signal processing, power electronics, power system and from other basic functions the simulink model with the ANFIS controller can be developed.The AC source is first rectified to get the smooth waveform along with the filter to reduce distortion and noise in the acquired waveform then the rectified output is given to the VSI as an input and the output is given to the ANFIS block for the selection of vectors. There frequency controlled voltage will be the output. The speed of induction is varied by using the controlled output of the inverter.

5.5 SIMULATION MODEL OF ANFIS BLOCK

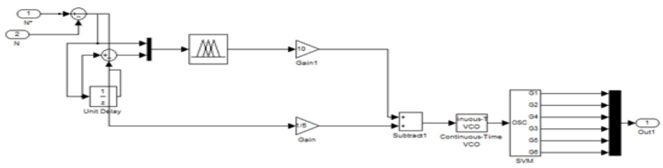
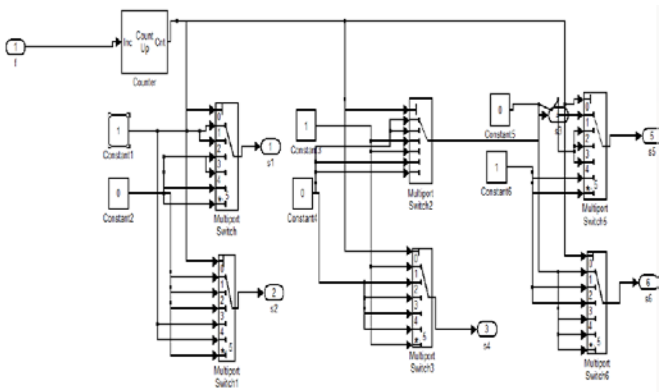


Figure 5.4 ANFIS Block for Controlling Three Phase Induction Motor

Simulation model of ANFIS block is shown in the figure.5.4. Here the neural output are trained with fuzzy logic. In this block the reference speed and running speed error is compared which is one input to the fuzzy controller other input is the unit gain to improve the value. The gain value is given to the subtract block there the required adjustment is made in the value of the voltage frequency. The numerical values of the voltage frequency is converted into the waveform by using oscillator then the oscillator output is given to the SVM block for the selection of vectors and thus speed of the induction motor is controlled by controlling the output of the inverter.

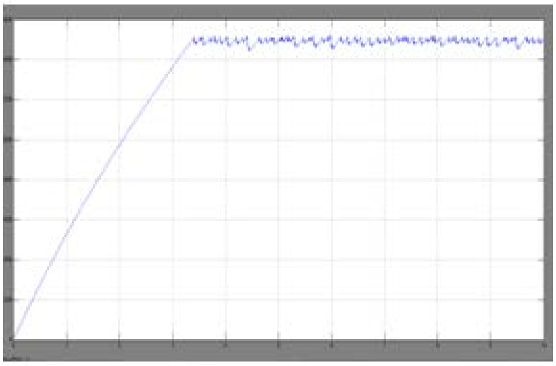
5.6 INTERNAL SPACE VECTOR MODULATION BLOCK



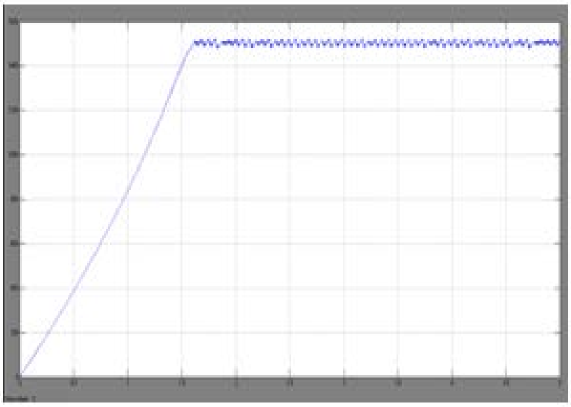
6. RESULTS

Speed waveforms obtained by ANN technique and ANFIS technique the are shown in the following Figures. By the waveforms it is clear that the speed control by ANFIS technique is better than the ANN technique also the rise time is decreased.

Speed Waveform of the Three Phase Induction Motor Drive with ANN with 1500 rpm



Speed Waveform of the Three Phase Induction Motor Drive with ANFIS With speed of 1500 rpm



7. CONCLUSION

Artificial neural network and Neuro – fuzzy are compared for various speeds. It is inferred with the speed wave forms obtained from the simulation, ANFIS based AI the rise time drastically decreases. According to the speed error from the set speed the inverter firing angles are generated, by this induction motor supply frequency is varied. Based on direct relation of induction motor speed and frequency of supplied voltage, the speed is increased and apparent system of ANFIS based motor control strategy has been proposed .With the result of simulation it is clear that ANFIS based system of motor reaches the rated speed in lesser time and has better performance than the ANN for the same operating condition of induction motor.

REFERENCES

[1] Ashok Kusagur et al (2009), ‘Modelling of Induction Motor & Control of Speed Using Hybrid Controller Technology’, Proc. Int. Journal of Theoretical Information & Technology (JATIT), Vol. 10, issue 2, pp. 117-126.

[2] Arulmozhiyal. R and Baskaran. K (2009), ‘Space Vector Pulse Width Modulation Based Speed Control of Induction Motor using Fuzzy PI Controller’, Proc. of the International Journal of Computer and Electrical Engg., Vol. 1, No. 1, pp. 98-103.

[3] Ernesto Araujo,(2008), ‘Improved Takagi-Sugeno Fuzzy Approach’, IEEE International Conference on Fuzzy Systems (FUZZ 2008), pp. 1154-1158.

[4] Haider A. F. Mohamed et al, (2008), ‘Fuzzy-SMC-PI Flux and Speed Control for Induction Motors’, Proc. of RAM-2008, pp. 325-328 .

[5] Kusagur. S. F. Kodad and Sankar Ram. B. V, (2009), ‘AI based design of a fuzzy logic scheme for speed control of induction motors using SVPWM technique’, Int. Jr. Comp. Sci. & Network Security, Vol. 9, No. 1, pp. 74 - 80.

[6] Menghal. P. M and Jaya Lakshmi. A (2013), ‘Artificial intelligence based induction motor drive’ Micheal Faraday IET India Summit, Kolkata , India , pp.208-212.

[7] Menghal.P.M ,and Jaya Laxmi. A “Artificial Intelligence Based Dynamic Simulation of Induction Motor Drives” IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE) ISSN: 2278-1676 Volume 3, Issue 5 (Nov. - Dec. 2012), pp 37-45.

[8] Mokhtar Zerikat and Sofiane CHekroun, (2008), ‘High performance speed tracking of IM using an adaptive fuzzy NN control’, Int. Jr. Sciences & Techniques of Auto. Contr. & Computer Engg., IJ-STA Special Issue, CEM, pp. 516-531.

[9] Rehman. H and Dhaouadi. R,(2008), ‘A fuzzy learning-sliding mode controller for direct field-oriented induction machinese’, Neuro-computing, Vol. 71, pp. 2693–2701.

[10] Chen. J. Y and Wong. C. C, (2000), ‘Implementation of the Takagi-Sugeno model-based fuzzy control using an adaptive gain controller’, IEE Proc. - Control Theory Appl., Vol. 147, No. 5, pp. 509 – 514.

[11] Farzan Rashidi, (2004), ‘Sensorless Speed Control of Induction Motor Derives Using a Robust and Adaptive Neuro-Fuzzy Based Intelligent Controller’, IEEE International Conference an Industrial Technology (ICIT), pp. 617-627.

[12] Ernesto Araujo, (2008), ‘Improved Takagi-Sugeno Fuzzy Approach’, IEEE International Conference on Fuzzy Systems (FUZZ 2008), pp. 1154-1158.

[13] Khiar. D, (2007), ‘Robust takagi-sugeno fuzzy control of a spark ignition engine’, Control Engg. Practice.