Comparative Analysis of Different Speed Controllers for a Vector Controlled Induction Motor Drive

Aeidapu Mahesh
Department of Electrical Engineering
NIT Kurukshetra

Pratibha Singh
Department of Electrical Engineering
NIT Kurukshetra

Abstract- **This paper presents the comparison of various speed control techniques for the vector controlled induction motor. Artificial Neural Network (ANN), Fuzzy Logic Controller (FLC) and conventional PI controller have been considered and compared. The data used for training the ANN speed controller has been taken from the conventional PI controller, with different values of proportional and integral gains. Based on the limits of speed error variations an FLC has been designed with a set of membership functions. And later the comparison has been done for all these three controllers. Since, the ANN and fuzzy logic controllers are adaptive in nature; the simulation results as obtained show that, they are producing a good dynamic response in contrast to the conventional PI controller at all reference speeds.**

Key words: *Vector control, speed controller, ANN, Fuzzy logic controller.*

I. INTRODUTION

Now-a-days the induction motor drives are the most sought drives in the industry, because of the high performance and robustness. The main objective of any drive is to follow the required speed trajectory irrespective of load variations, parameter changes due to physical operating conditions. So, for achieving good performance, the vector control or Field oriented Control (FOC) of induction motor drive is used. However the decoupling provided by the FOC is greatly influenced by the parameter variation [1].

Previously the traditional proportional and integral (PI) and proportional integral and differential (PID) controllers are used for the speed control of induction motor. However, the traditional PI & PID controllers are fixed gain controllers and the performance of these controllers is greatly sensitive to the change of the parameters. Thus, for improved performance of the drive, the parameters of the speed controller must be adaptive in nature. The model reference adaptive control (MARS) [2], sliding mode control (SMC) [3] and self-tuning PI controllers [4] are generally used instead of conventional controllers. Major drawback of these models is the requirement the exact mathematical model of the system. Whereas, getting the exact mathematical model is a tedious job, due to the dependence of machine parameters on temperature, unpredictable nature of load variation and the disturbances in the system. So, as a solution to the problems the artificial intelligent techniques namely, fuzzy logic and ANN can be used [5]-[7]. The main advantage of these techniques over the other methods is they don't require the accurate system mathematical model and as these techniques are based on simple linguistic rules or a set of data so; the non-linearity in the models can be easily handled.

Both the conventional and the artificial intelligent techniques have their own merits and demerits. The work presented in this paper gives the comparative analysis of the performance of induction motor drive with all the three speed controllers. The entire model has been developed and simulated on the Matlab/Simulink platform. The subsequent sections of the papers, the indirect vector control scheme, brief overview about the artificial intelligent techniques and simulation results, followed by a brief discussion on the results and conclusion are presented.

II. MATHEMATICAL MODEL OF THE INDUCTION MOTOR

The dynamic model of induction motor is generally expressed in two phase quantities, in order to achieve conceptual simplicity. The power invariance is taken as the key constraint for the equivalence for these transformations. The model of an induction machine is generally written in two reference frames; namely stator reference frame and rotor reference frame. These transformations include both voltage-current transformations and the flux linkage transformations [8].

The model in rotor reference frame is given by

$$\begin{bmatrix} V_{qs} \\ V_{ds} \\ V_{qr} \\ V_{dr} \end{bmatrix} = \begin{bmatrix} R_s + L_s p & \omega_s L_s & L_m p & \omega_s L_m \\ -\omega_s L_s & R_s + L_s p & -\omega_s L_m & L_m p \\ L_m p & \omega_{sl} L_m & R_r + L_r p & \omega_{sl} L_r \\ -\omega_{sl} L_m & L_m p & -\omega_{sl} L_r & R_r + L_r p \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix}$$

$$\begin{bmatrix} \lambda_{qs} \\ \lambda_{ds} \\ \lambda_{qr} \\ \lambda_{dr} \end{bmatrix} = \begin{bmatrix} L_s & 0 & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix}$$

Where, V_{ds}, V_{dr}, V_{qs}, V_{qr} are the stator and rotor q-d axes voltages, i_{ds}, i_{dr}, i_{qs}, i_{qr} are the stator and rotor q-d axes currents, λ_{ds}, λ_{dr}, λ_{qs}, λ_{qr} are the q-d axes flux linkages, R_r and R_s are the rotor and stator resistances, L_s and L_m are the stator and rotor self-inductances respectively, L_m is the mutual inductance, ω_s is the angular velocity of the d-q frame of axes, ω_{sl} is the slip angular velocity, p is the derivative operator.

III. STRATEGY OF INDIRECT VECTOR CONTROL SCHEME

Based on the mathematical model of the induction motor only, the indirect vector control strategy has been built. The inputs to the scheme are the flux and the rotor speed and from which the scheme produces the output commands as the flux producing component of the current and the torque producing component of the current and the slip angle. The slip angle command value Θ_{sl}*, which is obtained by integrating the speed command value ω_{sl}* and field angle Θ_f, which is a combination of command slip angle and rotor angle.

The equations (1)-(4) will form the mathematical model of the indirect vector control scheme, by considering the flux producing component of rotor current at a constant value by taking $i_{dr}=0$. With this assumption, the torque producing component can be independently controlled.

$$i_{ds}^* = \left(\frac{2}{3}\right)\left(\frac{2}{P}\right)\frac{T_e^*}{\lambda_r^*}\frac{L_r^*}{L_m^*} \quad (1)$$

$$i_{qs}^* = (1 + T_r p)\left[\frac{\lambda_r}{L_m^*}\right] \quad (2)$$

$$\omega_{sl}^* = \left[\frac{L_m^*}{T_r}\right]\left[\frac{i_r}{\lambda_r}\right] \quad (3)$$

$$T_e = \left(\frac{3}{2}\right)\left(\frac{P}{2}\right)\frac{L_m}{L_r}\left(\lambda_{dr}\, i_{qs}\right) \quad (4)$$

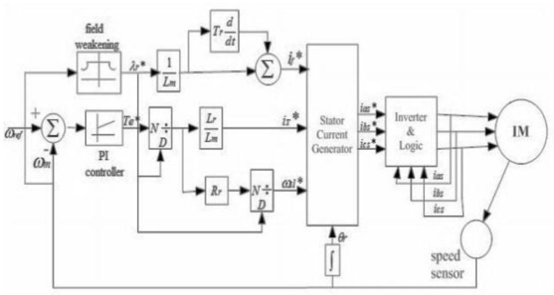

Fig 1. Typical indirect vector control scheme.

IV. ANN SPEED CONTROLLER

The neural network, is one of the most used artificial intelligent technique, because of its adaptability in non-linear systems. The neural network resembles human brain in two ways, one the knowledge acquired is through the learning process and the inter neuron synaptic weights, which are used to store the information [9].

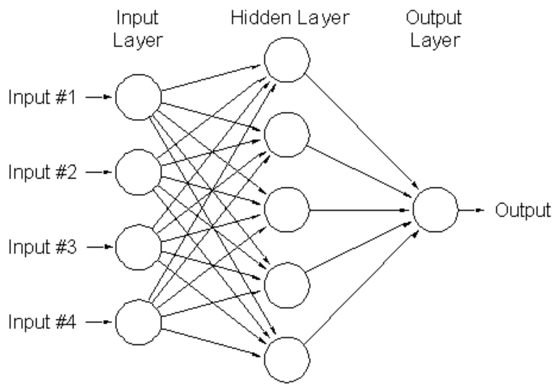

Fig 2. Structure of a 3-layer Feed forward ANN

Structure of a three layered feed-forward neural network, has been shown in the Fig 2. Here, the first layer is generally called as the input layer and the last layer is the output layer. The intermediate layers are called as hidden layers. All these layers are inter-connected by the links. Each link in the network will have its own weight; the input to the neuron in the i^{th} layer will be the outputs of the all connected neurons multiplied by their respective weights as per the equation (5).

$$x_i = \sum_{j=1}^{n} w_{ij}.y_j \quad (5)$$

Where, y_j is the output of the previous neuron, w_{ij} is the weight of the link connecting j_{th} and i_{th} neuron and the n is the number of neurons in the j_{th} layer. If the sum of these products exceeds the value of the activation function of the present neuron, then it gets fired. Here, the main objective when we are trying to apply the neural network to any system is the weights of the links to each neuron must be decided in order to achieve the desired value of output. The process of assigning the weights to the links is called training or learning. The learning process is mainly two types; “supervised” and “unsupervised”. When, the training exercise is being carried out with the desired set of both inputs and outputs, it is called as supervised. In this case the error between the desired and outputs will be propagated back, so as to update the weights of the links. This process is widely known as back-propagation.

For the proposed ANN speed controller, the reference speed (ω_r^*), motor speed (ω_r) are given as the inputs and the command torque value (T_e^*) is taken as the output.

The architecture of the proposed system is a three layered feed-forward network, consisting of 2 neurons in input layer, 5 neurons in the hidden layer and 1 neuron in the output layer. The back-propagation algorithm has been used for training. The data for training the neural network has been taken from the data of conventional PI controller with different values of K_p and K_i.

V. Fuzzy Logic Controller

The fuzzy logic controller is one of the most versatile controllers, because of the simplicity and applicability to most of the systems. Unlike the other techniques, the fuzzy logic basically works on the linguistic variables, which represents a specific range of input. The inputs and outputs are converted to linguistic sets, by means of a process called “fuzzification”. Then, the output is obtained by forming a set of if-then rules between the input and output. These set of rules are called as rule base. The final stage of getting crisp output from the fuzzy output is called “defuzzification”.

The controller proposed in this work takes the speed error (ΔE) as the input and produces the command value of torque (T_e^*) as the output. The input has been divided into five membership functions, namely; negative high (NH), negative medium (NM), negative small (NS), zero (ZE), positive small (PS), positive medium (PM) and positive high (PH). The output is also divided into three linguistic variables namely; negative (NE), medium (ME) and positive (PE) as shown in fig 3&4.

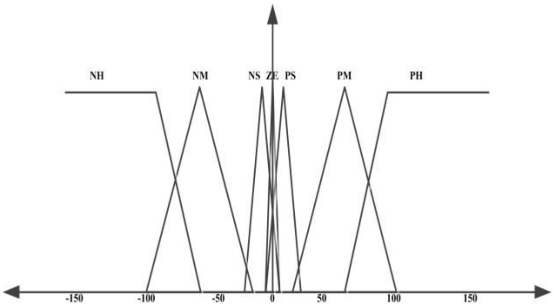

Fig 3. Membership functions for speed error

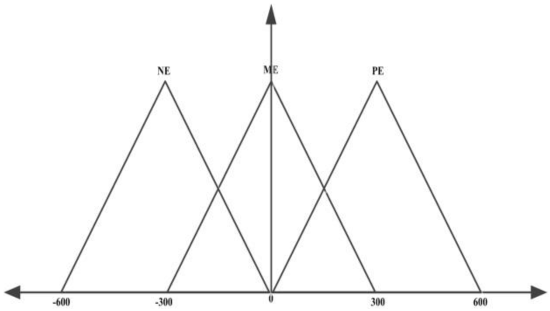

Fig 4. Output membership functions for T_e^*

The input membership functions are the combinations of both triangular and trapezoidal membership functions but the output has only triangular membership functions. This particular strategy covers all possible inputs and reduces the execution time. All the input/output membership functions and the rule base have been obtained based on the trial and error method for the optimum performance of the drive. The rule base has been presented in the Table 1.

TABLE 1
FUZZY RULE BASE

1. If the speed error $\Delta\omega_n$ is NH then T_e^* is NE
2. If the speed error $\Delta\omega_n$ is NM then T_e^* is NE
3. If the speed error $\Delta\omega_n$ is NS then T_e^* is NE
4. If the speed error $\Delta\omega_n$ is ZE then T_e^* is ME
5. If the speed error $\Delta\omega_n$ is PS then T_e^* is PE
6. If the speed error $\Delta\omega_n$ is PM then T_e^* is PE
7. If the speed error $\Delta\omega_n$ is PH then T_e^* is PE

VI. RESULTS AND DISCUSSIONS

Several simulations have been performed to evaluate the performance of the drive under various operating conditions with different speed controllers proposed. For fair comparison of the conventional PI controller with the Fuzzy and ANN controllers it has been tuned to the rated conditions. To show the dynamic performance, the simulation has been carried out by applying sudden load to the motor and different speed command values.

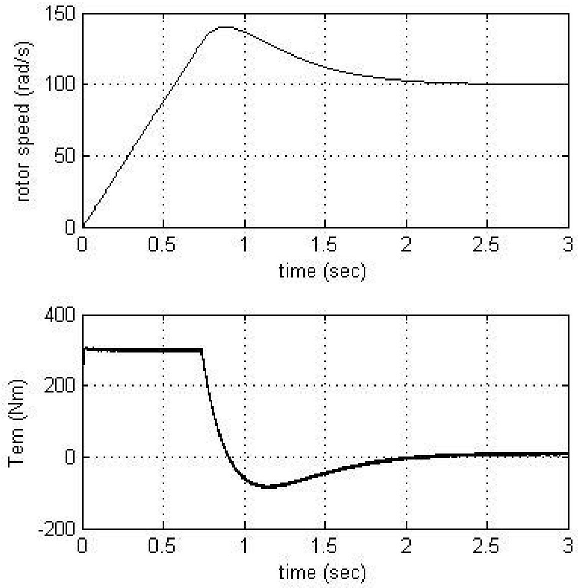

Fig 5. Speed and torque waveforms on no load with 100 *rad/s* ref speed

The simulations with the conventional PI controller without and with load in fig. 5. The speed and torque responses of the drive with 100 rad/s speed command value and the fig. 6 shows the same with a load is being applied at 4 sec.

The responses of the drive with the ANN tuned speed controller has been shown in the fig 7&8. The response shown with and without load clearly shows, there is a lot of improvement in terms of dynamic performance of the drive, like almost no overshoot and the settling time of the speed curve is very less when compared to the conventional PI controller. But, the limitation of the ANN controller is the response of the drive is steeper.

The responses with the fuzzy logic controller have been shown in fig 9&10. The fuzzy controller is also shows a great improvement in the performance. There is considerably no overshoot and the settling time is a little more compared to the ANN controller. But, the response with the fuzzy logic controller is smoother compared to the ANN controller.

When a load has been applied on a fuzzy logic controlled drive, the speed is slightly reduced. This is due to the fact that, the fuzzy logic gives the output based on a set of input membership functions, actually represents a

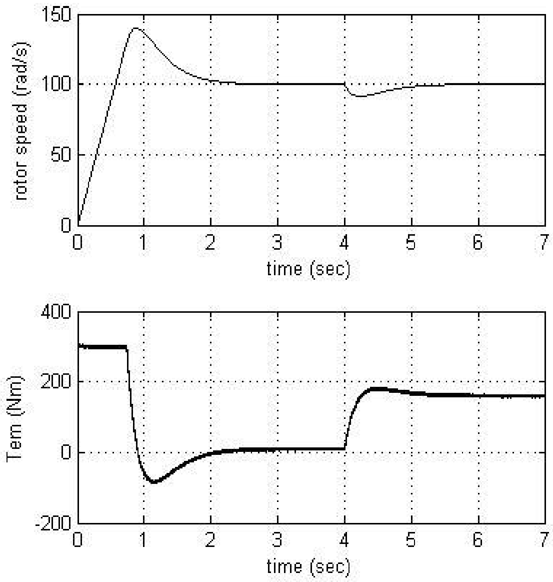

Fig .6 Speed and torque waveforms with an applied step load of 150 *Nm*

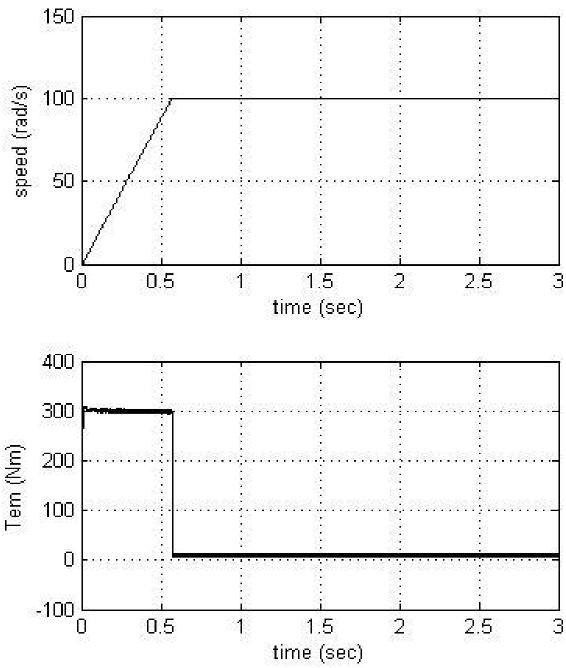

Fig. 7 Rotor speed and Torque responses with ANN speed controller with 100 *rad/s* ref speed

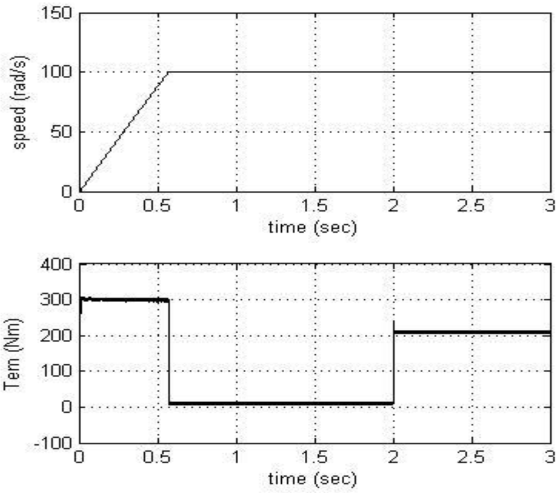

Fig. 8 Speed and Torque responses with an applied step load of 200 *Nm*

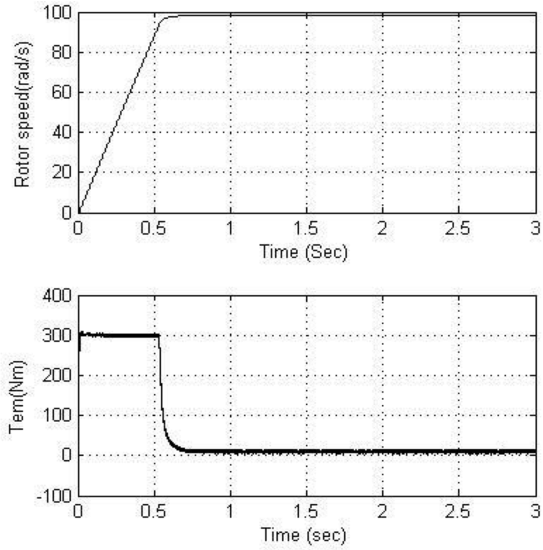

Fig .9 Speed and torque waveforms on no load with 100 *rad/s* ref speed

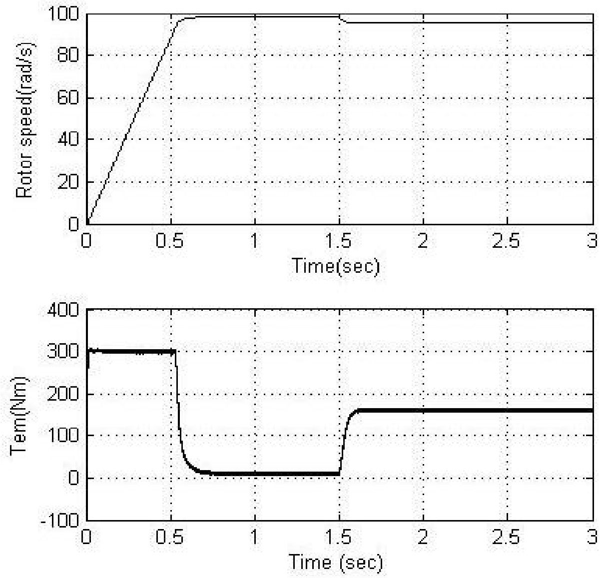

Fig .10 Speed and torque responses with a step load of 150 *Nm*

range of input values. As in this case, the input is speed error, when a load applied it causes the motor to reduce the speed, but the reduction is quite small, so the input linguistic variables are unable to recognize the change as they are defined over a range. For solving this problem, the number of input membership functions need to be increased. But, this unnecessarily increases the computational burden if the reduction in change of speed is within the acceptable limits. So, one has to look after the tradeoff between the computational time and the accuracy in the response for effective implementation of the drive.

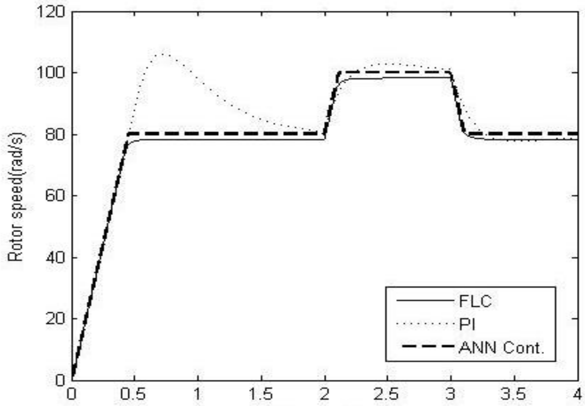

Fig .11 Speed responses for different speed command values for ANN, FLC and PI controllers

The comparison between the all three speed controllers has been presented in fig 11. Here, the initial reference speed is set as 80 *rad/s* and the a positive step change of 20 *rad/s* is applied at 2 sec and later a negative step change of 20 *rad/s* is applied at 3 sec.

As shown in the fig .11, the PI speed controller is producing an output with an overshoot and more settling time, whereas the fuzzy logic controller is producing a smooth output without overshoot but there is a small steady state error. But, the conventional PI controller is always easy to design. The ANN controller is accurately following the speed command value without any error. But for implementing the ANN controller with the accuracy you need to have large data of input and output variables covering all the operating conditions. Each and every technique has its own merits and demerits, the best method need to be chosen based on the performance required.

VII. CONCLUSIONS

In this paper, a comparative analysis of three speed controllers for a vector controlled induction motor drive has been presented. The drive has been simulated using all three speed controllers. Comparative analysis of simulation results on a motor using conventional PI controller, ANN and controller and Fuzzy logic controller indicates the superiority of ANN controller over the other controllers.

VIII. REFERENCES

[1] . M. Nasir Uddin, Tawfik S. Radwan and M. Azizur Rahman, "performances of Fuzzy-Logic-Based Indirect Vector Control for Induction Motor Drive", *IEEE Trans. on Ind. Appli.* , VOL. 38, NO. 5, Oct. 2002, pp. 1219-1225.

[2] . H. Sugimoto and S. Tamai, "Secondary resistance identification of an induction motor applied model reference adaptive system and its characteristics," *IEEE Trans. Ind. Applicat.*, vol. IA-23, pp. 296–303, Mar./Apr. 1987.

[3] . C. Y. Won and B. K. Bose, "An induction motor servo system with improved sliding mode control," in *Proc. IEEE IECON'92*, pp. 60–66.

[4] . J. C. Hung, "Practical industrial control techniques," in *Proc. IEEE IECON'94*, pp. 7–14.

[5] . Aeidapu Mahesh and Balwinder Singh, "Vector control of induction motor using ANN and partical swarm optimization", *IJETAE*, Vol. 2, No. 9, Sept. 2012, pp. 480-485.

[6] . Yang Liyong, Li Yinghong; Chen Yaai and Li Zhengxi, "A novel fuzzy logic controller for indirect vector control induction motor drive", *WCICA* 2008, pp. 24 – 28.

[7] . Jadhav, S.V, Kirankumar J and Chaudhari, B.N., "ANN based intelligent control of Induction Motor drive with Space Vector Modulated DTC", *IEEE Int. Conf. on Power Electronics, Drives and Energy Systems (PEDES)*, Dec 2012, pp. 1-6.

[8] . R. Krishnan, "Electric Motor Drives Modeling, Analysis and Control", *PHI learning private Ltd*., 2001.

[9] . Simon Haykin, "Neural networks, a comprehensive foundation", 2nd edition, Pearson Prentice hall, 2005.