# Development and implementation of a simplified self-tuned neuro-fuzzy-based IM drive

V.PRAKASH[1]
*PG STUDENT.EEE DEPARTMENT,SVPCET,PUTTUR*
G.KRISHNA DEEKSHIT[2]
*ASSISTANT PROFESSOR,EEE DEPARTMENT,SVPCET,PUTTUR*

**Abstract:** A novel simplified self-tuned neuro–fuzzy controller(NFC) for speed control of an induction motor (IM) drive is presented in this paper. The proposed NFC combines fuzzy logic and a four-layer neural network structure. Only speed error is employed as input to the NFC so that the computational burden of the NFC is reduced and it becomes suitable for real-time industrial drive applications. Based on the knowledge of back-propagation algorithm, an unsupervised self-tuning method is developed to adjust membership functions and weights of the proposed NFC so that the performance will be similar to that of the conventional two-input NFC. The complete drive incorporating the proposed self-tuned NFC is experimentally implemented using a digital-signal-processor board DS-1104 for a laboratory 1/3-hp motor. The effectiveness of the proposed NFC-based vector control of IM drive is tested in both simulation and experiment at different operating conditions. Comparative results show that the simplification of the proposed NFC does not decrease the system performance as compared to the conventional NFC. In order to prove the superiority of the proposed simplified NFC, the performances of the proposed NFC are also compared to those obtained by a conventional proportional–integral controller.

**Index Terms:** Back propagation (BP), digital signal processor (DSP), indirect field-oriented control (FOC), induction motor (IM), neuro–fuzzy control, real-time implementation, self-tuning.

## I. INTRODUCTION

INDUCTION motors (IMs) have been widely used as workhorse in the industry over the years due to its low cost, and simple and robust construction. However, the control of IM is complex due to its nonlinear nature, and the parameters change with operating conditions. Traditionally, the conven-tional fixed-gain proportional–integral (PI) and PI–derivative (PID) controllers and their adaptive versions have been widelyused for motor drives. However, the fixed-gain and adaptive controllers often suffer from chattering in the steady state, parameter variations, and load disturbances. Over the last two decades, researchers have been working to apply intelligent algorithms for motor drives due to some of their advantages as compared to the conventional PI, PID controllers and their adaptive versions. The main advantages are that the designs of these controllers do not depend on accurate system mathematical model and their performances are robust.

In this paper, a neuro–fuzzy controller (NFC) is considered because of the limitations of both fuzzy logic control (FLC) and artificial neural network (ANN) controllers. A fuzzy controller used for speed control of motor drive has asymmetric member-ship functions which need much more manuaadjusting by trial and error if optimized performance is wanted .

On the other hand, it is extremely tough to create a serial of training data for ANN that can handle all the operating modes . The NFCs, which overcome the disadvantages of FLC and ANN controllers, have been utilized by authors and other re-searchers for motor drive applications. Despite many advantages of intelligent controllers, the industry has been still reluctant to apply these controllers for commercial drives due to high computational burden imposed by large number of membership functions, weights, and rules, particularly on self-tuning condition High computational burden leads to low sampling frequency, which is not sufficient for real-time implementation. In , only weights were tuned online, but the membership functions were fixed to keep the computational burden at reasonable level. The membership functions were adjusted in simulation by trial-and-error procedure. Moreover, a high-torque ripple was observed due to the low sample rate for the conventional two-input NFC. A fast processor may be used to implement such high computational intelligent algorithms, but the high cost of the fast processor is another concern for the industry.

Conventional NFCs usually utilize two inputs $\omega$ and $\omega^{\cdot}$ (speed error and acceleration, respectively), which lead to a large number of membership functions and rules. The adoption of $\omega^{\cdot}$ can improve the controller's robustness . How-ever, the difficulty of measuring fast and precise acceleration deteriorates this ability and even makes utilization of acceleration useless. Therefore, in order to reduce the computational burden, a simplified NFC with one input, three membership functions, and a four-layer structure is proposed in this paper. A supervised self-tuning method is developed based on the knowledge of back-propagation (BP) algorithms and vector control requirements. The main task of the tuning method is to adjust the parameters of the membership functions and weights in order to minimize the square of the speed error between actual and reference values.

The performance of the proposed simplified NFC-based IM drive is tested in both simulation and experiment. It is found that the proposed NFC does not decrease the system performance as compared to the conventional NFC. Moreover, the proposed NFC is found superior to the conventional PI controller.

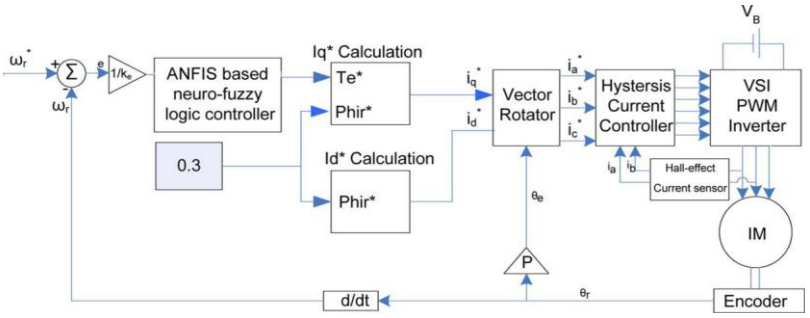

Fig. 1. Block diagram of the proposed simplified NFC-based IM driveand vector control requirements. The main task of the tuning method is to adjust the parameters of the membership functions and weights in order to minimize the square of the speed error between actual and reference values.

The performance of the proposed simplified NFC-based IM drive is tested in both simulation and experiment. It is found that the proposed NFC does not decrease the system performance as compared to the conventional NFC. Moreover, the proposed NFC is found superior to the conventional PI controller.

## II. MOTOR DYNAMICS AND CONTROL STRUCTURE

### *A. IM Dynamics*

The mathematical model of a three-phase Y-connected squirrel-cage IM in a $d^e - q^e$ synchronously rotating reference frame is described in , shown at the bottom of the page, where $v_{ds}{}^e$ and $v_{qs}{}^e$ are the *d*, *q*-axis stator voltages, $i^e{}_{ds}$ and $i^e{}_{qs}$ are the *d*, *q*-axis stator currents, $i^e{}_{dr}$ and $i^e{}_{qr}$ are the *d*, *q*-axis rotor currents, $R_s$ and $R_r$ are the stator and rotor resistances per phase, and $L_s$ and $L_r$ are the self-inductances of the stator and rotor, respectively; $L_m$ is the mutual or magnetizing inductance; $\omega_e$ is the speed of the rotating magnetic field; $\omega_r$ is the rotor speed; *P* is the number of poles; *p* is the differential operator (*d/dt*); $T_e$ is the developed electromagnetic torque; $T_L$is the load torque; $J_m$is the rotor inertia; $B_m$is the rotordamping coefficient; and $\theta_r$ is the rotor position. The motor parameters are given in the Appendix.

The key feature of the field-oriented control (FOC) is to keep the magnetizing current at a constant rated value by setting $i^e{}_{qr} = 0$. Thus, the torque-producing current component $i^e{}_{qs}$can be adjusted according to the torque demand. With thisassumption, the mathematical formulations can be rewritten as

$$\omega_{sl} = \frac{R_r}{L_r} \frac{i^e_{qs}}{i^e_{ds}} \qquad (5)$$

$$i^e_{ds} = \frac{\lambda^e_{dr}}{L_r} \qquad (6)$$

$$T_e - \frac{3}{2} \frac{P}{2} \frac{L_m}{L_r} \lambda^e_{dr} i^e_{qs} \qquad (7)$$

where $\omega_{sl}$ is the slip speed and $\lambda^e{}_{dr}$ is the $d$-axis rotor flux linkage. Equations are used to simulate the whole drive system. The schematic diagram of the proposed NFC-based indirect FOC of IM is shown in Fig. 1. The basic configuration of the drive system consists of an IM fed by a current-controlled voltage source inverter (VSI). The normalized speed error

$\omega$%is processed by the NFC to generate the reference torque $T_e{}^*(n)$. The command current$i^*{}_q(n)$is calculated from (7) asfollows:

$$i_q^*(n) = T_e^*(n)\frac{2}{3}\frac{2}{P}\frac{L_r}{L_m}\frac{1}{\lambda_{dr}^*}. \qquad (8)$$

Currents i∗q and i∗d are transformed into i∗a, i∗b, and i∗c by inverse Park’s transformation. The phase command currents i∗a, i∗b, and i∗c are then compared with the corresponding actual currents ia, ib, and ic to generate pulsewidth-modulation (PWM) logic signals, which are used to trigger the power semiconductor switches of the three-phase inverter. The inverter produces the actual voltages to run the motor.

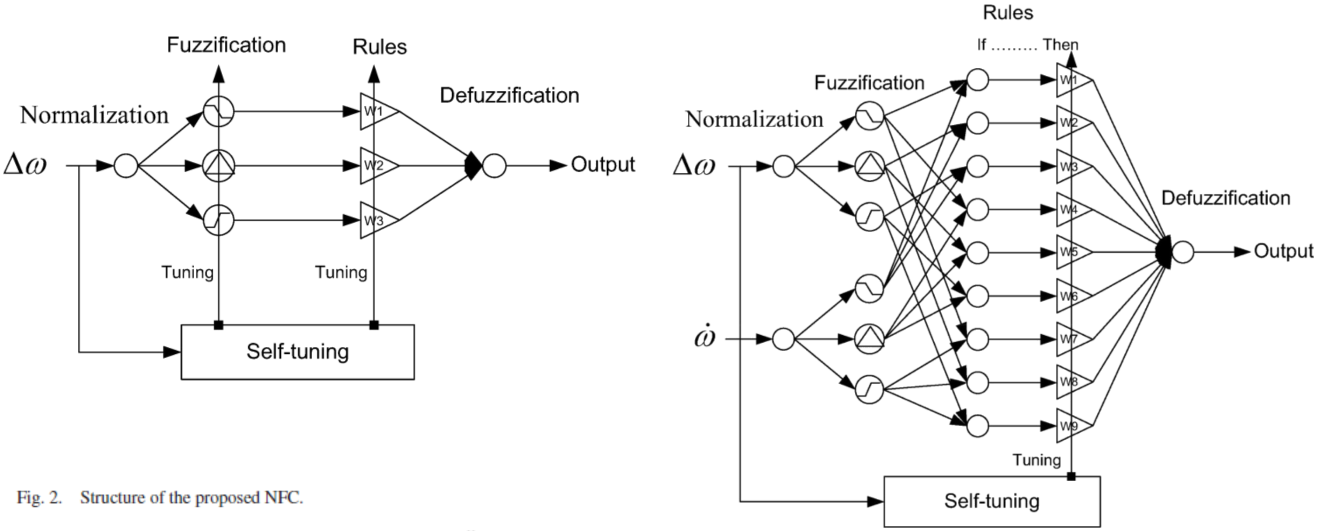

Fig. 2. Structure of the proposed NFC.

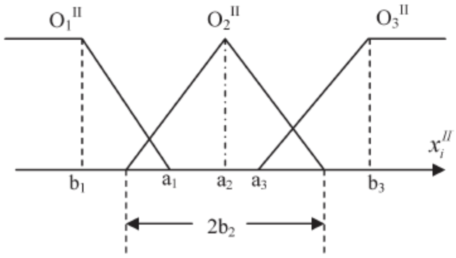

Fig. 3. Membership functions for input.

## III. DESIGN OF SPECIFIC NFCS

### A. Proposed NFC

The proposed NFC incorporates fuzzy logic and a learning algorithm with a four-layer ANN structure, as depicted in Fig. 2. The learning algorithm modifies the NFC to closely match the desired system performance. The detailed discussions on different layers of the NFC are given in the following.

*Input Layer:* The input of the proposed NFC is the normalized speed error, which is given by

$$O^I = \frac{\omega^* - \omega}{\omega^*} * 100\%$$

where$\omega$ is the measured speed, $\omega^*$ is the command speed, and $I$ denotes the first layer.

Fuzzification layer: In order to get fuzzy number from input ($x^{II}_i$ ), three membership functions $O_1^{II}$ , $O_2^{II}$ , and $O_3^{II}$ are used, which are shown in Fig. 3. The three nodes in fuzzification layer of NFC shown in Fig. 2 represent these three membershipfunctions . Here O stands for output, superscript indicates the layer number and subscripts indicate the mode numbers. The linear triangular and trapezoidal functions so that the computational burden is low as compared to any exponential in
Funtions.

As per Fig. 3, the equations representing the three member-ship functions can be written as follows:

$$O_1^{II} = \begin{cases} 1, & x_i^{II} \le b_1 \\ \frac{x_i^{II}-a_1}{b_1-a_1}, & b_1 < x_i^{II} < a_1 \\ 0, & x_i^{II} \ge a_1 \end{cases} \qquad (10)$$

$$O_2^{II} = \begin{cases} 0, & \left|x_i^{II}\right| \ge b_2 \\ 1 - \frac{x_i^{II}-a_2}{b_2}, & \left|x_i^{II}\right| < b_2 \end{cases} \qquad (11)$$

$$O_3^{II} = \begin{cases} 0, & x_i^{II} \le a_3 \\ \frac{x_i^{II}-a_3}{b_3-a_3}, & a_3 < x_i^{II} < b_3 \\ 1, & x_i^{II} \ge b_3 \end{cases} \qquad (12)$$

Fig. 4. Structure of a conventional NFC.

where $x^{II}{}_i$ is the input of the second layer, which is the output of the first layer. It is considered that $a_2 = 0$ so that the member-ship functions become symmetrical and it also further reduces the computational burden. Thus, the membership functions $O_1{}^{II}$, $O_2{}^{II}$, and $O_3{}^{II}$represent negative, zero, and positive speederrors, respectively.

*Rule Layer:* No "AND" logic is needed in the rule layer sincethere is only one input in the input layer. The node equations in the rule layer are specified as

$$E = \frac{1}{2}r^2 = \frac{1}{2}(\omega^* - \omega)^2. \qquad (15)$$

where $x^{VI}{}_i$ is the input of the fourth layer, which is same as the output of the third layer
*B. Conventional NFC*

$$y_u = O_i^{VI} = \frac{\sum x_i^{VI}}{\sum O_j^{II}} = \frac{\sum O_i^{III}}{\sum O_j^{II}} \qquad (14)$$

**B. Conventional NFC**

In order to compare the performance of the proposed simplifiedNFC, a conventional two-input NFC is also designed, as shown in Fig. 4. The normalized speed error and its derivative

NFC, a conventional two-input NFC is also designed, as shown in Fig. 4. The normalized speed error and its derivative

Since it is impossible to determine or calculate the desired NFC output $i_{qs}{}^e$ and find training data offline covering all operating conditions, a kind of unsupervised online self-tuning method is developed based on BP algorithm . Instead of using the desired controller output $i_{qs}{}^e$ as target, a reinforcement signal (*r*), which assesses the performance of the controller and evaluates the current state of the system, is employed to guide our control action into changing in the right direction as well as produce the desired response. The task of NFC is to modify its parameters so that the objective function of the reinforcement signal in is decreased.

The objective function to be minimized is defined by

$$E = \frac{1}{2}r^2 = \frac{1}{2}(\omega^* - \omega)^2. \qquad (15)$$

Hence the learning rules can be derived as follows

$$a_i(n+1) = a_i(n) - \eta_{a_i}\frac{\partial E}{\partial a_i} \qquad (16)$$

$$b_i(n+1) = b_i(n) - \eta_{b_i}\frac{\partial E}{\partial b_i} \qquad (17)$$

$$w_j(n+1) = w_j(n) - \eta_{w_j}\frac{\partial E}{\partial w_j} \qquad (18)$$

Where $\eta_{a1}, \eta_{b1}$, *and* $\eta_{c1}$ are the learning rates of the corresponding parameter.Thederivate can be found by chain rules as

$$\frac{\partial E}{\partial a_i} = \frac{\partial E}{\partial r}\frac{\partial r}{\partial \omega}\frac{\partial \omega}{\partial y_u}\frac{\partial y_u}{\partial O_i^{II}}\frac{\partial O_i^{II}}{\partial a_i} \qquad (19)$$

$$\frac{\partial E}{\partial b_i} = \frac{\partial E}{\partial r}\frac{\partial r}{\partial \omega}\frac{\partial \omega}{\partial y_u}\frac{\partial y_u}{\partial O_i^{II}}\frac{\partial O_i^{II}}{\partial b_i} \qquad (20)$$

$$\frac{\partial E}{\partial w_j} = \frac{\partial E}{\partial r}\frac{\partial r}{\partial \omega}\frac{\partial \omega}{\partial y_u}\frac{\partial y_u}{\partial w_j} \qquad (21)$$

Where the common parts of (19)-(20) are

$$\frac{\partial E}{\partial r} = r = \omega^* - \omega \qquad (22)$$

$$\frac{\partial r}{\partial \omega} = -1 \qquad (23)$$

$$\frac{\partial \omega}{\partial y_u} = J(t). \qquad (24)$$

In(24).The Jacobean matrix J(t) cannot be fount easily. In FOC the IM system can be viewed as a single- input single- output system then the J(t) can be estimated as a constant value Kj> 0.

From the update rules can be determined as follows.

$$a_1(n+1) = a_1(n) - \eta_{a_1}K_J r(n)\frac{w_1(n)}{\sum O_j^{II}}\frac{1 - O_1^{II}(n)}{b_1(n) - a_1(n)} \qquad (25)$$

$$b_1(n+1) = b_1(n) - \eta_{b_1}K_J r(n)\frac{w_1(n)}{\sum O_j^{II}}\frac{O_1^{II}(n)}{b_1(n) - a_1(n)} \qquad (26)$$

$$b_2(n+1) = b_2(n) + \eta_{b2}K_J r(n)\frac{w_2(n)}{\sum O_j^{II}}\frac{1 - O_2^{II}(n)}{b_2(n)} \qquad (27)$$

$$a_3(n+1) = a_3(n) - \eta_{a_3}K_J r(n)\frac{w_3(n)}{\sum O_j^{II}}\frac{1 - O_3^{II}(n)}{b_3(n) - a_3(n)} \qquad (28)$$

$$b_3(n+1) = b_3(n) - \eta_{b_3}K_J r(n)\frac{w_3(n)}{\sum O_j^{II}}\frac{O_3^{II}(n)}{b_3(n) - a_3(n)} \qquad (29)$$

$$w_j(n+1) = w_j(n) + \eta_w K_j r(n)\frac{O_i^{II}(n)}{\sum O_i^{II}}. \qquad (30)$$

Based on the aforementioned update rules, the following stepsare employed for tuning the parameters of $a_1$, $a_3$, $b_1$, $b_2$, $b_3$,and $w_j$ .

Step 1) First an initial set of fuzzy logic rules
and initial values of a1,a3, b1,b2,b3, and
are selectedStep 2) The normalized speed error is calculated

which is input to the NFC

Step 3) Fuzzy reasoning is performed for input data. The membership values $O_i^{II}$ are then calculated by using .

Step 4) Tuning of the weights wj of the consequent parts is performed by using (25).

Step 5) Tuning of a1, a3, b1,b2, and b3 is done by substituting the tuning real number wj obtained in step 4) , the measured reinforcement signal r. and the membership valves $O_i^{II}$ into .

Step 6) Repeat from step 3).

The tuning rate for weights is chosen to be $\eta_{a1}$ = 0.1 and the tuning rates for the membership functions is chosen to be $\eta_{a1} = \eta_{i1}$ = 0.008.The small values of tuning rates are chosen so that there will be a smooth transition from one state to another.

UDDIN *et al.*: DEVELOPMENT AND IMPLEMENTATION OF A

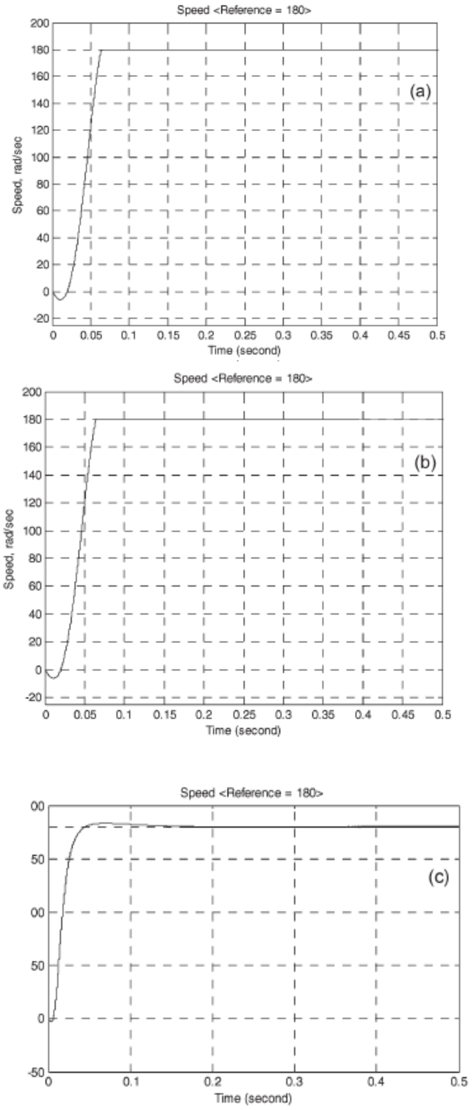

Fig.6 simulated starting speed responses of the drives (a) Proposed NFC (b) Two input NFC (c) PI controller.

## V EXPERIMENTAL SETUP

The proposed simplified self- tuned NFC-Based vector control of IM drive system has been implemented in real time using the digital –signal-processor (DSP) board DS1104(28). This board is mainly based on 64-b floating – point MPC8240

Processorwith PPC603e core. The photograph of the experimental system system is shown in Fig.  The actual motor currents are measured by the Hall-effect sensors and fed back to the DSP board through the A/D channels. Rotor position is sensed by an optical incremental encoder of 1000-line resolution and is fed back to the DSP board through the encoder interface. The outputs of the DSP board are PWM logic signals, which are sent to the VSI switches through driver circuitry. The test IM is coupled to a dc generator, which is used as a load to the IM.

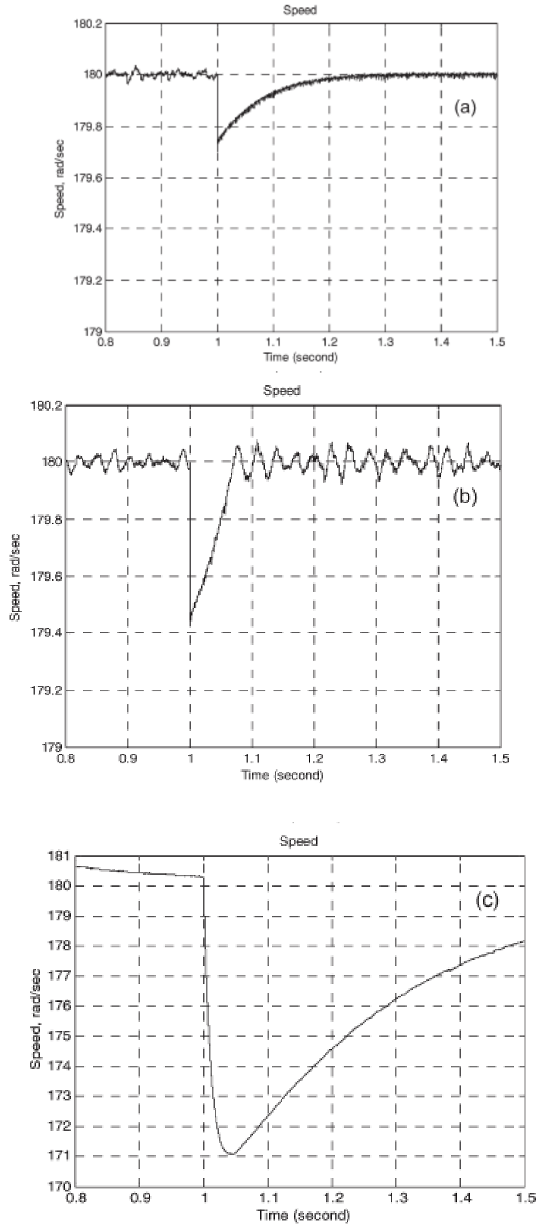

Fig. 7. Simulated speed responses of the IM drive at a step increase in load. (a) Proposed NFC. (b) Conventional two-input NFC. (c) PI controller.

The NFC and self-tuning algorithm are implemented through developing a real-time Simulink model. Then, the model is compiled and downloaded to the DSP board utilizing ControlDesk software and real-time workshop. Since the proposed NFC has a simple structure, the highest sampling frequency can reach up to 14.3 kHz. In this paper, the sampling frequency is set to be 10 kHz so that the experiment becomes closer to practical application. For comparison purposes, a conventional two-input NFC-based system is also developed and experimentally implemented. The sampling frequency for the conventional NFC is found to be 5 kHz. For comparison purposes, a PI-controller-based system is also developed and experimentally implemented. After trial and error, the proportional gain $K_p$ and the integral gain $K_i$ are selected as 0.1 and 0.02, respectively, so that there is no steady-state error and the settling time,

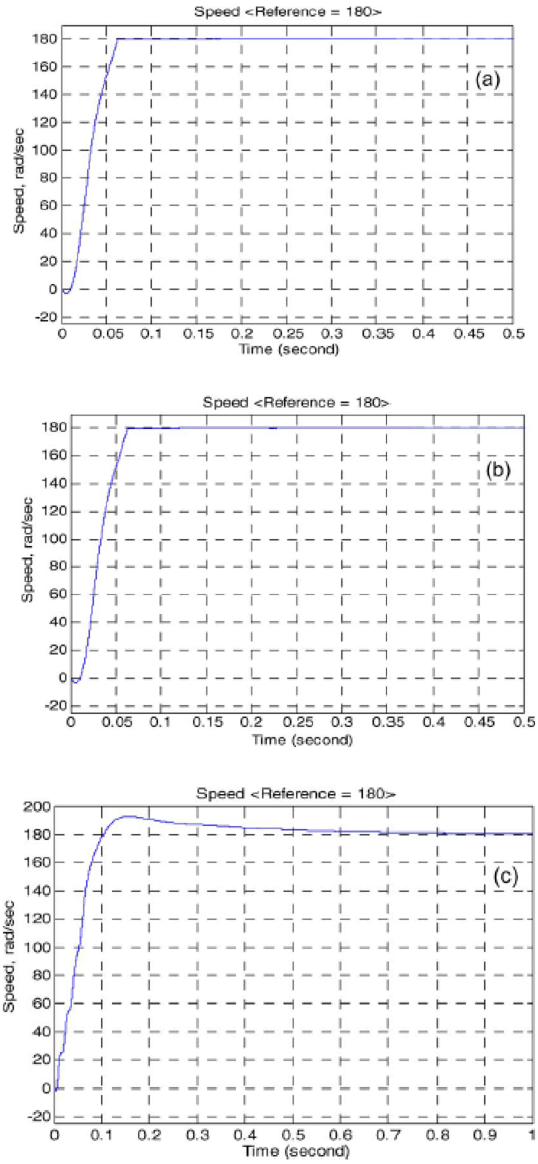

Fig.8. Simulated speed responses of the IM drive with doubled rotor resistance. (a) Proposed NFC. (b) Conventional two-input NFC. (c) PI controller

Overshoot, and undershoot can be comparable to those of the NFCs. If the PI controller is made critically damped, it became too sluggish, and the response time is not even comparable to that of the NFCs.

## VI. RESULTS AND DISCUSSION

### A. Simulation Results

The performance of the proposed simplified NFC-based IM drive is investigated in simulation using Matlab/Simulink at different operating conditions .

Fig. 6(a)–(c) shows the simulated starting performances of the drive at full load with the proposed NFC, conventional two-input NFC, and PI controllers, respectively. It is clearly seen from these figures that the performance of the proposed simplified low computational NFC is similar to that of the conventional NFC and, at the same time, it is superior to that of the conventional PI controller in terms of overshoot and settling time. Fig. 7(a)–(c) shows the zoom-in view of the speed

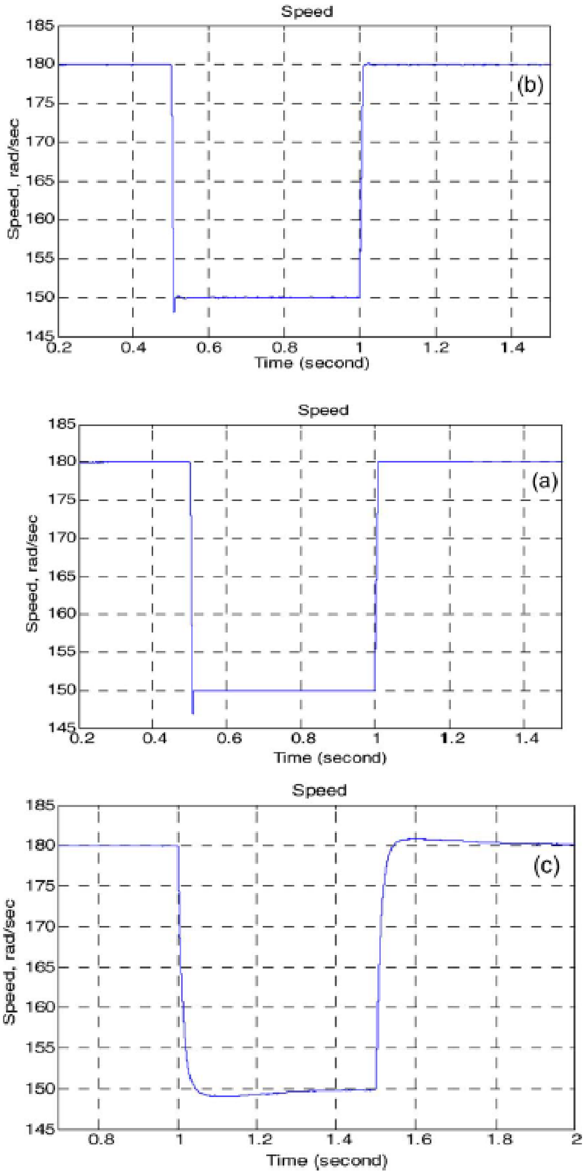

Fig.7. Simulated speed responses of the IM drive at a step increase in load.(a) Proposed NFC. (b) Conventional two-input NFC. (c) PI controller

The NFC and self-tuning algorithm are implemented through developing a real-time Simulink model. Then, the model is compiled and downloaded to the DSP board utilizing ControlDesk software and real-time workshop. Since the proposed NFC has a simple structure, the highest sampling frequency can reach up to 14.3 kHz. In this paper, the sampling frequency is set to be 10 kHz so that the experiment becomes closer to practical application. For comparison purposes, a conventional two-input NFC-based system is also developed and experimentally implemented. The sampling frequency for the conventional NFC is found to be 5 kHz. For comparison purposes, a PI-controller-based system is also developed and experimentally implemented. After trial and error, the proportional gain $K_p$ and the integral gain $K_i$ are selected as 0.1 and 0.02, respectively, so that there is no steady-state error and the settling time,

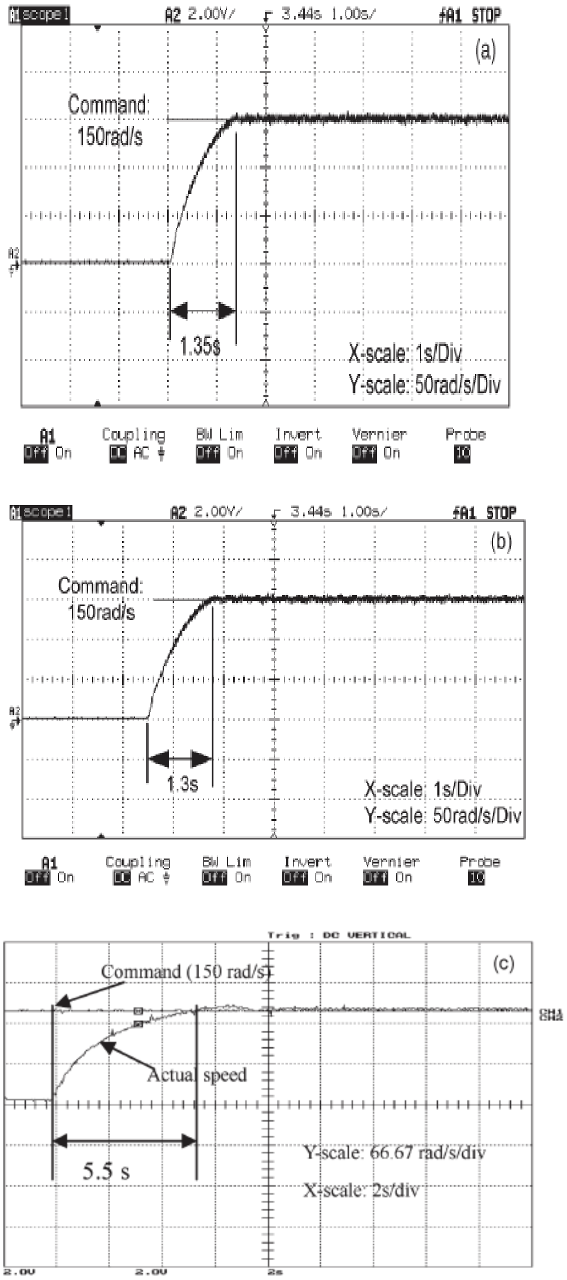

Fig10. Experimental starting performances of the IM drive for (a) proposed NFC, (b) conventional two-input NFC, and (c) PI controller.

controller as the PI controller takes longer time to reach the steady state. Based upon tests, it is evident that the proposed NFC does not decrease system performance significantly as compared to the conventional two-input NFC [13]. In addition, the simplified NFC provides superior performance as compared to the conventional PI controller.

### B. Experimental Results

Fig. 10(a)–(c) shows the experimental starting responses of the IM drive with the proposed NFC, conventional two-input NFC, and PI controller, respectively. It is seen from these figures that, in real time, the performance of the proposed simplified NFC is almost similar to that of the conventional NFC and superior to that of the PI controller in the context of settling time and overshoot. Thus, the experimental result validates the simulation. For further experimental results, the PI controller is not considered as it is already established in

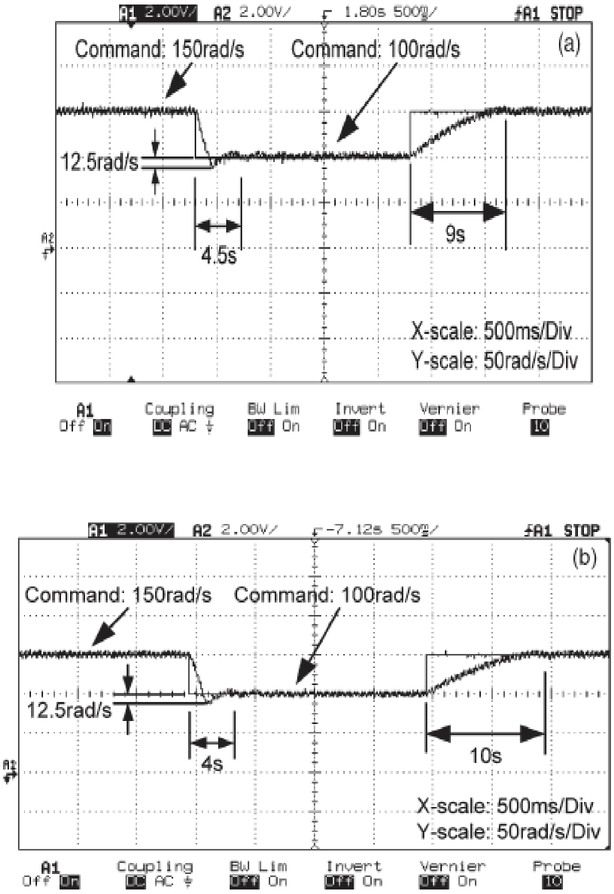

Fig.11. Experimental performances of the IM drive due to step change in command speed for (a) proposed NFC and (b) conventional two-input NFC

sample experimental and extensive simulation results that the NFC provides better response as compared to the PI controller. Next, the proposed simplified NFC is only compared with the conventional NFC. The experimental speed responses with step increase and decrease in command speed are shown in Fig. 11(a) and (b) for the proposed and conventional NFCs, respectively. In system can handle the step in-crease/decrease in command speed and load change smoothly. Thus, the proposed NFC shows fairly same performance as the conventional two-input NFC while reducing the computational burden significantly. Furthermore, the robustness of the proposed simplified NFC-based IM drive with parameter variation is also tested in experiment and shown in Fig. 13. The motor inertia was increased by coupling the tested motor with another motor of higher inertia value. It is found that the motor can follow the command speed smoothly with doubled inertia, but the settling time is increased due to higher inertia as compared to Fig. 10(a).

## VII. CONCLUSION

In this paper, a novel and simplified low computational online self-tuning NFC-based speed control of IM drive has been developed and experimentally implemented for a laboratory 1/3-hp motor. In the proposed NFC, conditions. The proposed controller can also be applied to other types of motors of different sizes only by adjusting the tuning rates. The comparison of the proposed NFC with

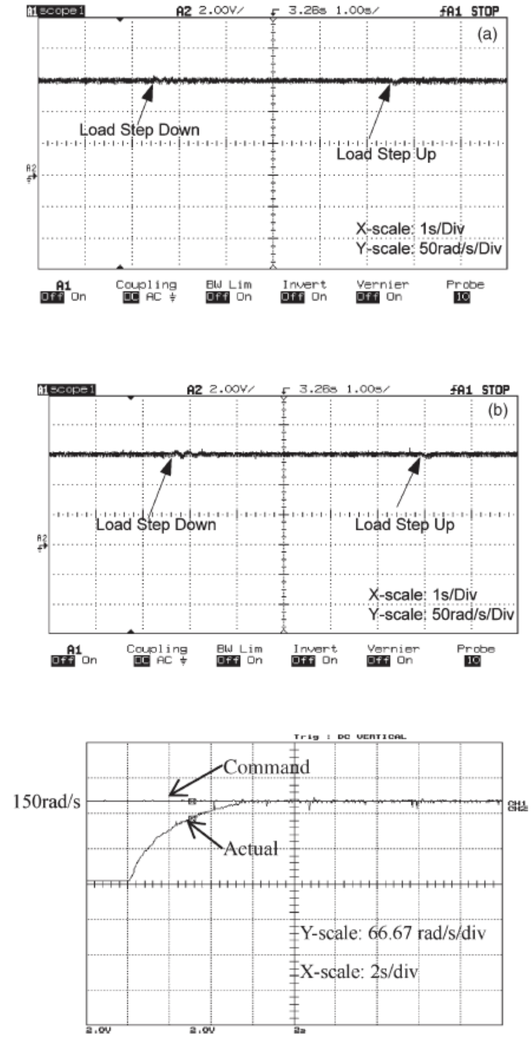

Fig.13. Experimental performance of the proposed simplified NFC-based IM drive with doubled inertia

conventional two-input NFC [13] and PI controllers has also been presented in both simulation and experiment. It is found from the results that the proposed simplified NFC exhibits almost the same performance of the conventional two-input NFC while reducing the computational burden significantly. The maximum sampling frequencies for the proposed NFC and conventional NFC have been found as 14.3 and 5 kHz, respectively, with the same DSP (DS1104) board. Moreover, the performance of the proposed NFC has been found superior to that of the conventional PI-controller-based IM drive. The proposed simplified self-tuned NFC-based IM drive system is found robust and could be a suitable candidate for real-time implementation of high-performance industrial drives.

## APPENDIX
## IM PARAMETERS (REFERRED TO STATOR SIDE)

| | |
|---|---|
| Power | 250W; |
| Stator Resistance | 6.5 Ω; |
| Rotor resistance | 3.4 Ω; |
| Number of pole pairs | 2; |
| Stator inductance | 0.0103H; |
| Rotor inductance | 0.0154H; |
| Mutual inductance | 0.2655H; |
| Inertia | 0.0012Kg m |
| Rated speed | 1725r/min. |

## REFERENCES

[1] G. W. Chang, G. Espinosa-Perez, E. Mendes, and R. Ortega, “Tuning rules for the PI gains of field-oriented controllers of induction motors,” *IEEETrans. Ind. Electron.*, vol. 47, no. 3, pp. 592–602, Jun. 2000.

[2] M. N. Uddin, T. S. Radwan, and M. A. Rahman, “Performances of fuzzy-logic-based indirect vector control for induction motor drive,” *IEEETrans. Ind. Appl.*, vol. 38, no. 5, pp. 1219–1225, Sep./Oct. 2002.

[3] E. C. Shin, T. S. Park, W. H. Oh, and J. Y. Yoo, “A design method of PI controller for an induction motor with parameter variation,” in *Proc. IEEEIECON*, 2003, vol. 1, pp. 408–413.

[4] A. Miloudi and A. Draou, “Variable gain PI controller design for speed control and rotor resistance estimation of an indirect vector con-trolled induction machine drive,” in *Proc. IEEE IECON*, 2002, vol. 1,pp323–328.

[5] R. Krishnan and A. S. Bharadwaj, “A review of parameter sensitivity and adaptation in indirect vector controlled induction motor drive systems,” *IEEE Trans. Power Electron.*, vol. 6, no. 4, pp. 695–703, Oct. 1991.

[6] C. M. Liaw, J. B. Wang, and Y. C. Chang, “A fuzzy adapted field-oriented mechanism for induction motor drive,” *IEEE Trans. Energy Convers.*, vol. 11, no. 1, pp. 76–83, Mar. 1996.

[7] C.-M. Lin and C.-F. Hsu, “Adaptive fuzzy sliding-mode control for induction servomotor systems,” *IEEE Trans. Energy Convers.*, vol. 19, no. 2,pp362–368, Jun. 2004.

[8] T. C. Huang and M. A. El-Sharkawi, “High performance speed and position tracking of induction motors using multi-layer fuzzy control,” *IEEETrans. Energy Convers.*, vol. 11, no. 2, pp. 353–358, Jun. 1996.

[9] Y. Tang and L. Xu, “Fuzzy logic application for intelligent control of a variable speed drive,” *IEEE Trans. Energy Convers.*, vol. 9, no. 4, pp. 679– 685, Dec. 1994

[10] F. F. Cheng and S. N. Yeh, “Application of fuzzy logic in the speed control of AC servo system and an intelligent inverter,” *IEEE Trans. EnergyConvers.*, vol. 8, no. 2, pp. 312–318, Jun. 1993.

[11] T. Chen and T. T. Sheu, “Model reference neural network controller for induction motor speed control,” *IEEE Trans. Energy Convers.*, vol. 17, no. 2, pp. 157–163, Jun. 2002.

[12] F.-J. Lin and R.-J. Wai, “Adaptive fuzzy-neural-network control for induction spindle motor drive,” *IEEE Trans. Energy Convers.*, vol. 17, no. 4,pp 507–513, Dec. 2002.

[13] M. N. Uddin and H. Wen, “Development of a self-tuned neuro–fuzzy controller for induction motor drives,” *IEEE Trans. Ind. Appl.*, vol. 43, no. 4, pp. 1108–1116, Jul./Aug. 2007.

[14] M. N. Uddin, “An adaptive filter based torque ripple minimization of fuzzy logic controller for speed control of a PM synchronous motor,” *IEEETrans. Ind. Appl.*, vol. 47, no. 1, pp. 350–358, Jan./Feb. 2011.