

Adaptive Fuzzy-Artificial Neural Network Based Speed Controller of D.C Motor Drive

Mohammed Golam Sarwer, Md. Abdur Rafiq and B.C. Ghosh

Department of Electrical & Electronic Engineering
Khulna University of Engineering and Technology
Khulna-9203, Bangladesh.
E-mail: mdsaku@yahoo.com

ABSTRACT

This paper introduces a D.C. motor drive system with a fuzzy-artificial neural-network controller. First, a neural network-based architecture is described for fuzzy logic control. The characteristic rules and their membership functions of fuzzy systems are represented as the processing nodes in the neural network structure. Then, the fuzzy rules and input-output of the system are tuned by the supervised gradient descent learning algorithm. The performance of the proposed controller is evaluated under various operating conditions. The controller is shown to be robust, adaptive and capable of learning.

1. INTRODUCTION

In the past decade, nonlinear and adaptive control methods have been used extensively to control DC and brushless dc motor drives [1-4]. In these methods, the state estimation and parameter identification are based on and limited to linear models. As the model deviates from the physical system, the performance of the control degrades. Furthermore, the methods require the exact equations and precise numerical values pertinent to the system under study. Li and Lau[1] applied fuzzy logic to microprocessor-based servomotor controller, assuming a linear amplifier. They compared the fuzzy controlled system performance with PID controller. Fuzzy logic can be applied in feedback control of dc drive system [5-6] in which the non-linearity, parameter variation and load disturbance are considered to be compared with fuzzy logic control. A comparison of drive system performance with fuzzy logic and PI control is given in [6] that improves the machine performance by linearizing the motor performance as the load varies.

Unlike classical control strategies, fuzzy logic incorporates an alternative way of thinking. It allows modeling of complex systems via the use of a higher level of abstraction originated from accumulated knowledge and experience. Further,

fuzzy logic permits expression of the knowledge by subjective concepts, such as very large, moderate, and slightly deviated, which are mapped onto numeric ranges. A fuzzy logic controller uses fuzzy logic as a design methodology, which can be applied in developing nonlinear systems for embedded control. Other researchers have used artificial neural networks (ANN) to deal with nonlinearities and uncertainties of Brushless DC drives [7-9]. ANNs have many advantageous features that include parallel and distributed processing and efficient nonlinear mapping between inputs and outputs, without an exact knowledge of the system model. Additionally, rapidity and robustness are the most profound and interesting properties of ANNs, in comparison to the classical schemes. Nonetheless, no underlying knowledge of the system's dynamics is assumed. Rather such knowledge is treated as a black-box which, when presented with a given input, produces a given output. However, one of the drawbacks of using ANNs is that it is difficult for the user to understand or modify the network decision-making process. Horikaya S. et al [12] have proposed Fuzzy Neural Network principles to Fuzzy reasoning. Basically, it emulates a fuzzy logic controller. This type of Fuzzy controller emulation has the advantages that it can automatically identify fuzzy rules and fuzzy membership function for a problem.

In this paper, an online trained Fuzzy-Artificial Neural-Network Controller (FANNC) for a DC motor drive system is proposed. The proposed controller integrates the ideas of the fuzzy logic controller and neural network structure into an intelligent control system. An ANN-based structure is introduced for the fuzzy logic control. The nodes in the hidden layers perform as membership functions and fuzzy rules. Initially, the proposed controller is constructed from the fuzzy IF-THEN rules, which is based on the simple engineering knowledge concerning the controlled DC drive system. A learning mechanism is then used to update the parameters of the adaptive FANNC. The supervised gradient decent method, which uses a delta adaptive law is utilized to train the proposed controller online. An adaptive fuzzy learning control system is developed in section 3. The proposed method is tested under normal and with disturbance added conditions by simulations.

2. DYNAMIC MODEL OF A SEPARATELY EXCITED DC MOTOR

A separately excited dc motor has the simplest decoupled electromagnetic structure. A schematic diagram of the electrical network of the machine is shown in Fig.1.

The field excitation is normally kept constant to produce rated flux. The armature current is controlled to generate desired electromagnetic torque and the armature voltage is controlled for the load. Assuming constant field excitation the electrical circuit differential equation is written as

$$v = R_a i_a + L_a \frac{di_a}{dt} + k_b \omega_m . \quad (1)$$

Where the parameters and variables have usual significance.

The dynamics of the mechanical system is given by the following first order differential equation

$$T_e = K_t i_a = T_l + J \frac{d\omega_m}{dt} + K_b \omega_m . \quad (2)$$

Where the parameters & variables have usual significance.

Equations 1 and 2 are rearranged to obtain

$$\frac{di_a}{dt} = \frac{v}{L_a} - \frac{R_a}{L_a} i_a - \frac{K_b \omega_m}{L_a} , \quad (3)$$

$$\frac{d\omega_m}{dt} = \frac{K_t i_a}{J} - \frac{T_l}{J} - \left(\frac{B}{J}\right) \omega_m . \quad (4)$$

Under steady state condition the speed of the motor is given by

$$\Omega_m = \frac{V - I_a R_a}{K_b} , \quad (5)$$

$$\Omega_m = \frac{V}{K_b} - \frac{R_a T_e}{K_b K_t} , \quad (6)$$

Equation 6 indicates that controlling the armature input voltage effectively controls the speed of the motor.

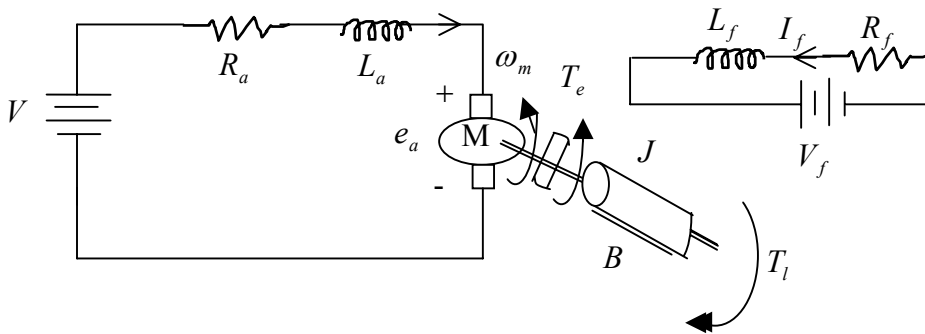


Fig. 1: Schematic diagram of separately excited dc motor.

3. ADAPTIVE FUZZY-NEURAL-NETWORK CONTROL SYSTEM

The developed control system, shown in Fig. 2, incorporates a FANNC and a learning algorithm.

The FANNC observes the behavior of the drive system and compares the actual performance to a

desired reference track. The learning algorithm modifies the FANNC to more closely match the desired system behavior. The parameter learning is based on the supervised gradient decent method. The inputs of the fuzzy neural controller are the speed error $e(k)$ and the change in speed error $\Delta e(k)=[e(k)-e(k-1)]$, respectively. The knowledge base for the fuzzy controller is generated from IF-THEN control rules. A set of such rules forms the “ rule base” which characterizes the manner in which a dynamic system is controlled. The FANNC described in

this paper consists of two inputs with five membership functions each, as shown in Fig. 3.

Hence, there are a total of 25 IF-THEN rules in the controller in Fig. 3. There are 10 triangular membership functions. The output of the FANNC, $\Delta u(k)$, is computed via the centroid defuzzification method. It should be noted that the FANNC is of the incremental type. Accordingly, the previous output, $u(k-1)$; is added to the output of the controller, $\Delta u(k)$. This supplies the ultimate output, $u(k)$, which is the input to the motor.

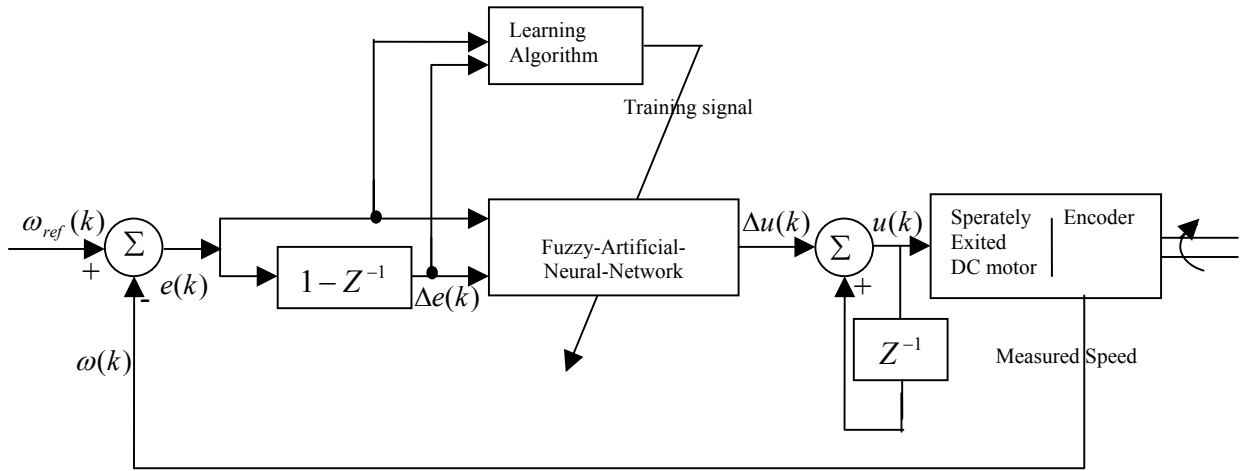


Fig. 2: Adaptive FANNC system.

4. ARCHITECTURE OF THE FANNC

The architecture of the proposed five-layer neural network-based fuzzy logic controller is shown in Fig.3. The two input nodes in layer 1 only transmit input signals to the next layer. Each node corresponds to one input variable. The input variables are the speed error and the change of speed error. For every node i in this layer, the input and the output of the network are represented, respectively, as

$$net_j^1 = X_i^1, \quad Y_i^1 = net_j^1. \quad (7)$$

Where X_i^1 represents the i th input to the node of layer 1.

The nodes in layer 2 are term nodes that act as membership functions to express the input/output fuzzy linguistic variables. The triangular activation functions $(\mu_e, \mu_{\Delta e})$, as shown in Figs. 4 and 5, are used to represent the membership

functions for speed error and change of speed error respectively.

$$net_j^2 = f_j^2(\mu_e \quad or \quad \mu_{\Delta e}),$$

$$Y_j^2 = f_j^2(net_j^2) = net_j^2. \quad (8)$$

The weights between the input and membership layer are assumed to be unity. The fuzzy sets defined for the input/output variables are positive large (PL), positive small (PS), zero (ZE), negative large (NL) and negative small (NS). Therefore, 10 and 25 nodes are included in layer 2 and 3, respectively, to indicate the input/output linguistic variables. Each node in layer 3 is denoted by $[\]$ which multiplies the incoming signal and outputs the result of the product. Consequently, each node of this layer is a rule node that represents one fuzzy control rule. In total, there are 25 nodes in layer 3 to form a fuzzy rule base for two linguistic input variables. The link of layer 3 defines the preconditions and the outcome of the rule nodes, respectively.

For each rule node, there are two fixed links from the input term nodes.

For the rule node

$$net_k^3 = \prod_j w_{jk}^3 X_j^3, \quad Y_k^3 = net_k^3. \quad (9)$$

Where X_j represents the j th input to the node of layer 3, and w_{jk} is the link that connects the output of the j th node in layer 2 with the input to the k th node in layer 3.

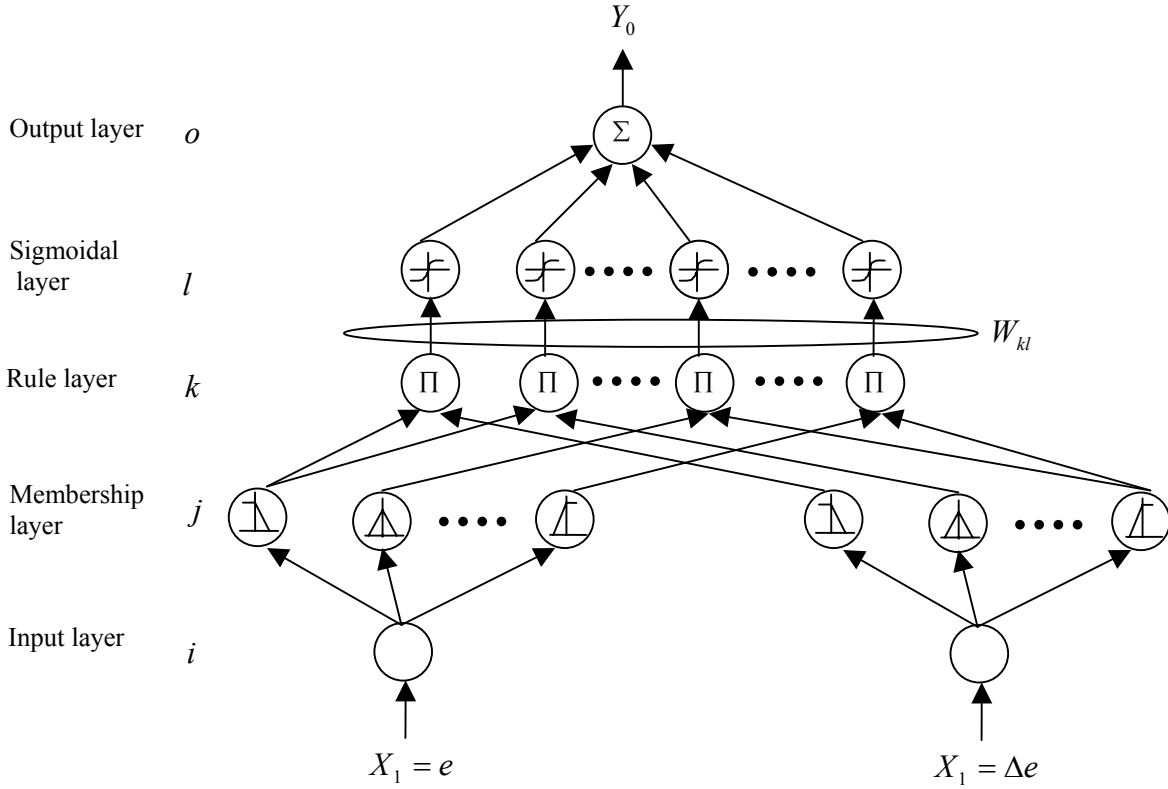


Fig. 3: Architecture of the neural fuzzy controller.

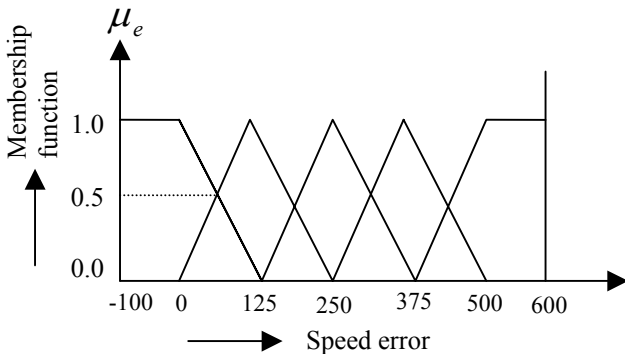


Fig. 4: Membership function for the speed error (μ_e).

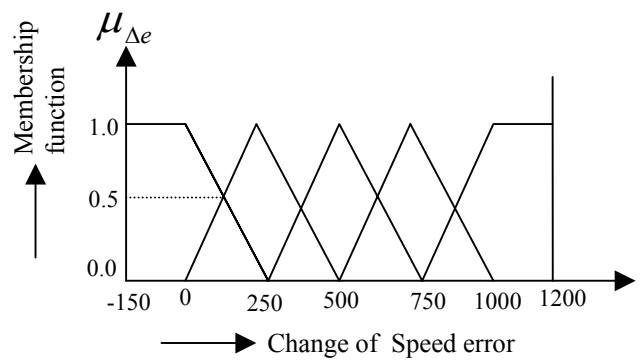


Fig. 5: Membership function for the change of speed error ($\mu_{\Delta e}$).

The weights between the input and membership layers are also assumed to be unity. The link of layer 4, surrounded by a dotted line, will be adjusted in response to varying control circumstances. The link weights w_{kl} represent the output action of the k th rule. Each node of layer 4 consists of nonlinear mappings, which is

sigmoidal function. The sigmoidal activation function imposes bounds on the signal and enhances stability. For the n th node of this layer, the input and output of the network are represented as

$$net_l^4 = Y_k, \quad Y_l^4 = \frac{2}{1 + \exp(-\gamma \cdot net_l^4)} - 1. \quad (10)$$

The output of the layer 5 is the output layer and acts as a defuzzifier. The single node Y_o in this layer is leveled \sum , and it sums all incoming signals to obtain the final inferred results

$$net_o^5 = \sum Y_l^4, Y_o = net_o^5. \quad (11)$$

The defuzzification aims at producing a nonfuzzy control action that best represents the possibility of distribution of an inferred fuzzy control action. The weighted average method [11], in which the fuzzy centroid constitutes the controller output, is used. Therefore, the change in control signal is given by

$$\Delta u = \frac{net_0^5}{\sum_k net_k^3}. \quad (12)$$

In summary, the FANNC controller can adjust the fuzzy control rules by modifying the link weights between layers k and l.

5. TRAINING SIGNAL ANALYSIS

The training signal analysis of the rotor speed tracking response is implemented to obtain some information regarding the drive system, prior to designing the FANNC. A step command is used as shown in Fig.6. As an example, the dotted line in Fig.6 depicts the exponential tracking response of the rotor speed. The solid line represents the response of the actual rotor speed. The speed error e and the change in speed error Δe are defined as

$$e(k) = w_{ref}(k) - w(k),$$

$$\Delta e(k) = e(k) - e(k - 1),$$

where $w_{ref}(k)$ and $w(k)$ are the respective responses of the desired reference track and actual rotor speed, at the k -th sampling interval.

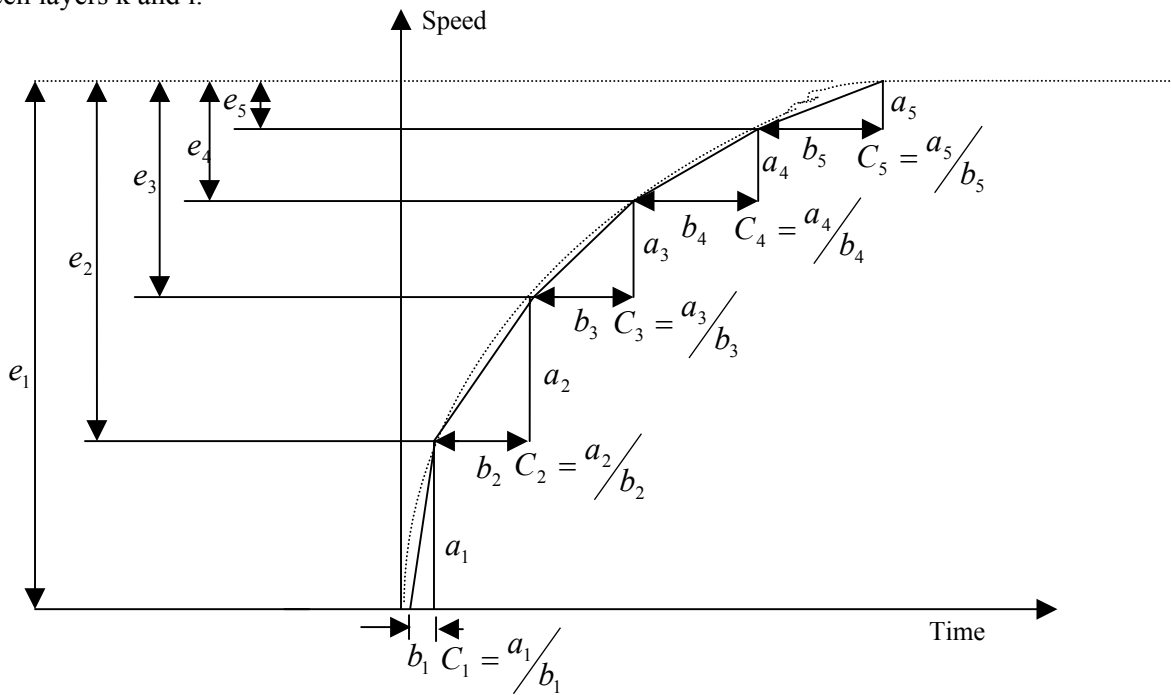


Fig. 6: Training signal analysis of the desired response.

In the most neural network applications, the error quantity is based on the actual and desired responses of the system. The output of the controller is a change in voltage. Since the system used is not modeled and has varying parameters, one does not have priori knowledge of the desired change in output. Such knowledge is essential for training any supervised neural network. The change in voltage cannot be accurately related to the speed error and, thus, the actual output of the

controller cannot be used as a training signal. Both the speed error and the associated changes are selected as appropriate training signals. A large error would require a large rate of change of the error with subsequent reduction in response time. Similarly, a small error would initiate a small rate of change of the error, thereby avoiding overshoot and possible oscillations. Thus the training signal used determines the desired response of the drive system.

Once the desired track is found, a set of rules to generate the training error can be developed by determining the derivative of the response track with respect to time. This may require a piecewise expression. With a suitable number of input regions, accurate results can be obtained.

The desired changes in speed errors $C(k)$, as follows:

$$\text{if } e(k-1) > e(k) \text{ then } C(k) = C_n.$$

After finding the desired value, the actual $\Delta e(k)$ is subtracted from $C(k)$ to give the training speed error $E(k)$, which will then be propagated using supervised gradient descent method. The training error is expressed as

$$E(k) = C(k) - \Delta e(k).$$

The weights between the rule layer and the sigmoidal layer are updated as follows:

$$\Delta w_{kl} = \gamma [E(k).net_k^3].$$

Where (γ) is the learning rate.

6. SIMULATION RESULT

The performance evaluation of the proposed speed controller was made by simulation on a digital computer environment. Parameters of the DC motor used for simulation are listed in Appendix. Equations (3) and (4) are solved simultaneously using Runge-Kutta –Gill method. Sampling time of the speed controller is 5 ms. To demonstrate the learning capability of the proposed controller the set speed of the dc motor was set 314.0 rad/sec. Figure 7 shows the controller output signal voltage and speed for this set speed. Some noise is introduced with motor speed. Fig. 8 shows the controller output signal voltage and speed for a random disturbance in speed measurement (output) of $\pm 3\%$. Time to reach the final settling value is increased but the output does not oscillate that indicates robustness of the control system.

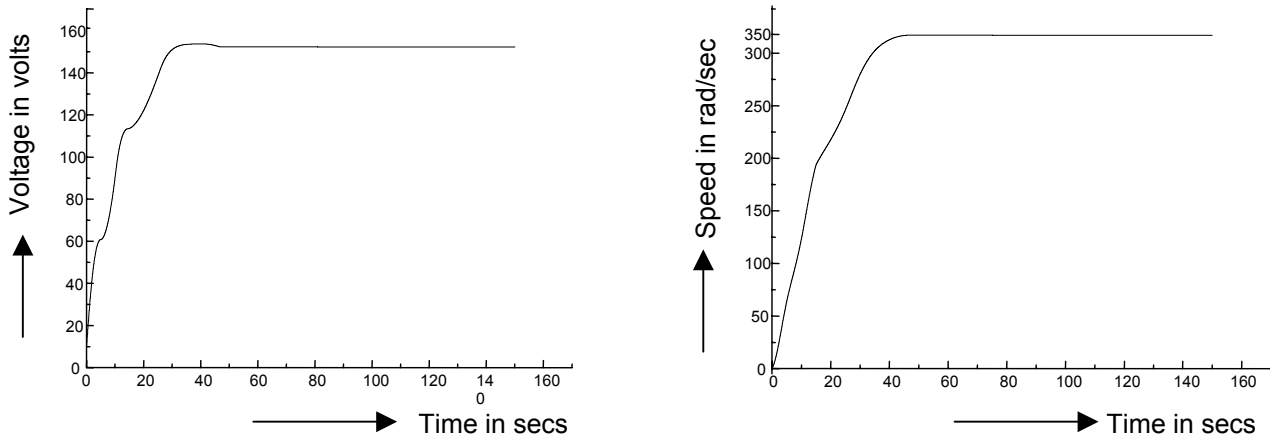


Fig. 7 : (a) Control signal voltage and (b) speed of the FANNC for set speed 314 rad/sec.

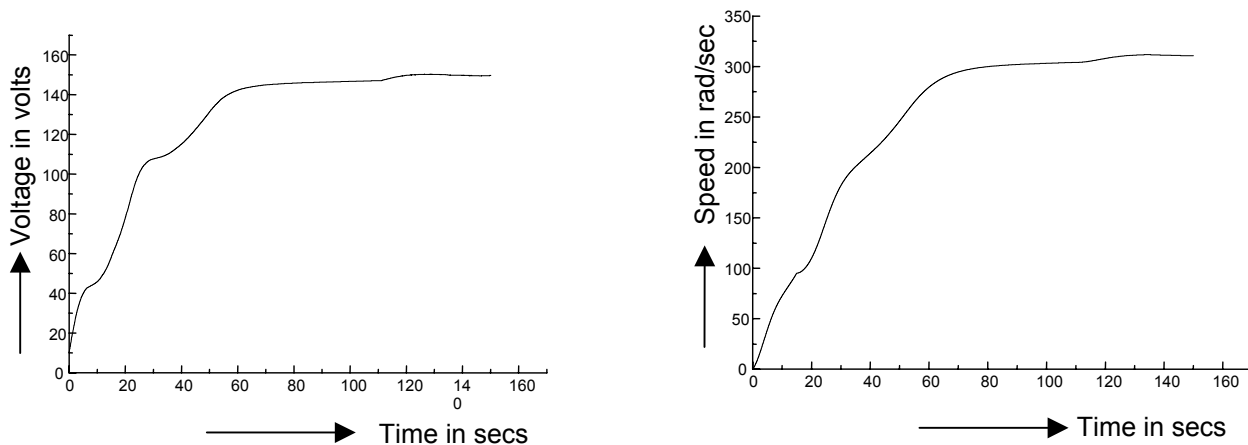


Fig. 8: (a) Control signal voltage and (b) speed of the FANNC for a disturbance in speed of $\pm 3\%$.

7. CONCLUSION

In this paper, an adaptive FANNC is successfully developed. The fuzzy logic-based control scheme is implemented by using a multi layer network. The network structure and the parameter learning are performed simultaneously and online in the fuzzy-neural-network. The proposed controller has several advantages, namely, simple structure and learning capability, robustness and high tracking performance and few fixed nodes at hidden layers.

APPENDIX

DC Motor parameters:

Armature resistance,	$R_a=7.56 \Omega$
Armature reactance,	$L_a= 0.055$ henry
Moment of inertia,	$J= 0.136$ Kg-m ²
Damping constant,	$B=.0002$ Kg-m ²
Load torque,	$T_l=1$ N-m.

REFERENCES

- [1] Li, Y.F. and C.C. Lau, "Development of Fuzzy Algorithms for servo systems", *IEEE control system Magazine*, April 1989.
- [2] N.Hemati, J. Thorp, and M.C. Leu, "Robust nonlinear control of brushless dc motors for direct-drive robotic applications," *IEEE Trans. Ind. Electron.*, vol. 3, pp. 562-575, June 1990.
- [3] B.Grcar, P.Cafuta, and M.Znidaric, "Practical robust stabilization of PAMC servo drive based on continuous variable structure control," *IEEE Trans. Energy Conversion*, vol.11, pp. 708-714, Dec 1996.
- [4] N.Matsui and H. Ohashi, "DSP-based adaptive control of a brushless motor," *IEEE Trans. Ind. Applicat.*, vol.28, pp.448-454, Mar./Apr. 1992.
- [5] Sousa G.C.D and B.K.Bose, "A fuzzy set theory based control of a phase controlled converter DC Machine Drive," *IEEE Trans. Industrial Application* vol. 30, pp 34-44 Jan/Feb 1994.
- [6] Rahman, M.M.H. Kamal, M.A.S. Islam M.M and Ghosh, B.C., "Fuzzy logic versus PI- control of Separately Excited DC Motor Drive-A comparison." *Journal of Electrical Engineering, IEB*, vol.27 No1, 1999, pp.29-36.
- [7] Y.Tang and L.Xu, "Fuzzy logic application for intelligent control of a variable speed drive," *IEEE Trans. Energy Conversion*, vol.9, pp.679-685, Dec. 1994.
- [8] A. Rubbai and R. Kotaru, "Neural net-based robust controller design for brushless DC motor drives," *IEEE Trans. Syst., Man, Cybern. C*, vol.29, pp.460-474, Aug 1999.
- [9] Ahmed Rubbai, Daniel Ricketts and M. David Kankam, "Development and Implementation of an adaptive fuzzy-neural-network controller for brushless DC drive," *IEEE Trans. of Industry Applications*, vol. 38, pp 441-446, March/april 2002.
- [10] Bernal, F.F., Cerredá, AG and Faure, R., "Model-Based loss minimization for DC and AC Vector Controlled Motor including core saturation," *IEEE Trans. Ind. Appl.*, vol 36, No 3, pp 755-763, May/June 2000.
- [11] T.J.Ross, *Fuzzy logic with engineering applications*. New york:Mcgraw-Hill, 1995.
- [12] Horikawa, S. et al, "Composition Methods of Fuzzy Neural Networks", *IEEE IECON conf. Proec.*, pp 1253-1258 1990.