ANALYSIS AND DESIGN OF CONTROL SYSTEMS USING ARTIFICIAL INTELLIGENCE

O.E.Gouda[1], A.A.Gafary[2], G. M.Amer[3] and O.M.Salim[3]*
[1]Electric Power and Mach., Faculty of Eng Cairo Univ., Egypt;
[2]Electric Power and Mach., Faculty of Eng Minia Univ., Egypt;
[3]Benha high institute of technology, Egypt
*E-mail: engomar@hotmail.com

Abstract: This paper introduces a comparative study between conventional speed controllers of dc motor and an artificial intelligent speed controller showing the advantages of artificial intelligent controller over the conventional ones.

All conventional controllers are implemented practically using a data acquisition card. Practical results are compared with simulation results for conventional controllers.

Design of artificial neural network controller is presented in this paper, the dc motor can be made to follow any arbitrarily selected speed, and the purpose is to achieve a fast and accurate speed controller.

Keywords: Speed Controllers, Artificial Neural Network, Data Acquisition Card, Choppers, PWM

1. Introduction

Control systems are an integral part of modern industrial society. Numerous applications are all around us, in many instances; the mathematical model of the plant is unknown or ill defined, leading to greater complexities in the design of the control system. It has been proposed that **intelligent control systems** give a better performance in such cases. Unlike conventional control techniques, intelligent controllers are based on artificial intelligence (AI) rather than on a plant model [1].

An artificial neural network (ANN) as a computing system is made up of a number of simple, and highly interconnected processing elements, which processes information by its dynamic state response to external inputs. In recent times the study of ANN models is gaining rapid and increasing importance because of their potential to offer solutions to some of problems, which have hitherto been intractable by standard serial computers in area of computer [2].

The suggested ANN controller in this paper proves that ANNs have a high degree of robustness, ability to learn, prepared to work with incomplete and unforeseen input data, high speed due to massive parallelism, it can trained rather than programmed; hence, their performance may improve with experience. ANNs capable of high-level function, such as adaptation or learning with or without supervision [1,2].

2. DC Motor Model

The dc motor is the obvious proving ground for advanced control algorithms in electric drives due to the stable and straight forward characteristics associated with it. It is also ideally suited for trajectory control applications. From a control systems point of view, the dc motor can be considered as a SISO plant, thereby eliminating the complications associated with multi-input drive systems [3,4].

2.1 Motor dynamics

The dc motor dynamics are given by the following two equations [5]

$$v_a = R_a i_a + L_a \frac{di_a}{dt} + K_b \omega \tag{1}$$

$$T_D = K_m . i_a = J \frac{d\omega}{dt} + F.\omega + T_L \tag{2}$$

Where,

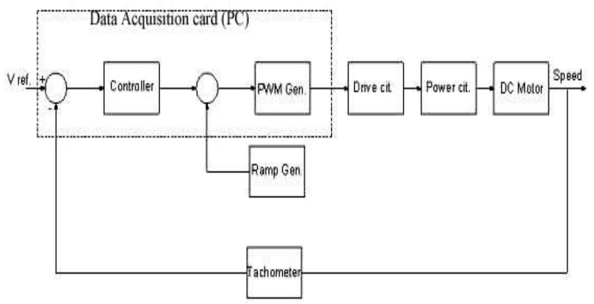
R_a	= armature resistance Ω
L_a	= armature inductance H
v_a	= armature input Source Voltage v
J	= rotor inertia Nm^2
F	= damping constant Nm
K_m, K_b	= torque & back emf constants NmA^{-1}
T_D	= developed torque Nm
T_L	= load torque Nm

In this paper a laboratory dc motor is used and all motor parameter are obtained experimentally by testing, and the typical values are given below

R_a	= 10 Ω
L_a	=0.06mH
V_a	= 50v.
F	=0.000128 Nm
J	=0.000192 Nm^2
K_m	=0.153 NmA^{-1}
K_b	=0.153 NmA^{-1}

2.2 Discrete time dc motor model

In order to obtain the discrete time



model for the dc motor some transformation should done first for equations (1), and (2). The sampling period for the transformed equations is (T=1ms) the discrete time domain model is as follows [1,6]

$$\omega(k+1) = 1.844\omega(k) - 0.846\omega(k-1) + 12.198 v_a(k) \qquad (3)$$

The no load step response of the dc motor model is shown in figure 1, all simulation results are obtained using MATLAB program, where the desired speed is 1500rpm.

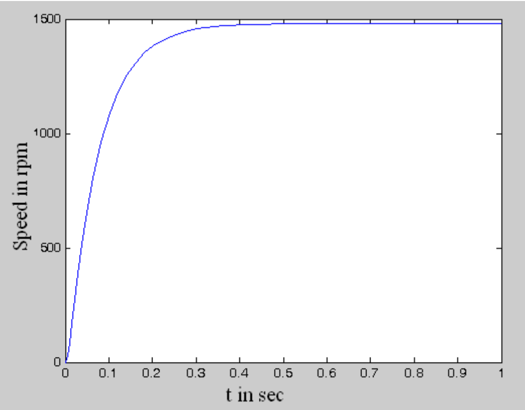


Figure 1 Simulation results of dc motor step response

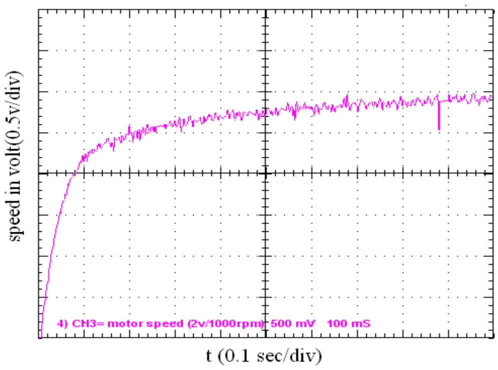


Figure 2 Practical results of dc motor step response

The open loop system response can be shown in figure 2, all practical results are obtained using a digital storage oscilloscope; all oscilloscope figures are stored while the horizontal setting is 0.1sec/div and the vertical setting is 0.5v/div, the rotor speed sensor is a tachometer with a constant of 2v/1000rpm.

2.3 Conventional controller design

In this paper, controllers' designs are done by root locus algorithm as it is considered the core of traditional control algorithms. Simulation results are drawn in the same graph with the practical results to evaluate the comparisons between theoretical and experimental results. In this paper four conventional controllers are presented and they are P-controller, PD-controller, PI-controller, and PID controller.

The proposed system block diagram is shown in figure 3.

Figure 3 Proposed system block diagram

In the proposed system the digital controller is achieved by the data acquisition card fitted in the PC which is programmed by C++, and it is used to read the reference voltage and the feedback signal given by the tachometer, then it is used to generate the PWM pulses to the power circuit which is a two quadrant dc chopper for the purpose of speed control and regeneration.

2.4 Conventional controllers Results discussion

For the compensated system with P-controller the simulation result is shown in figure 4, and the practical result is shown in figure 5.

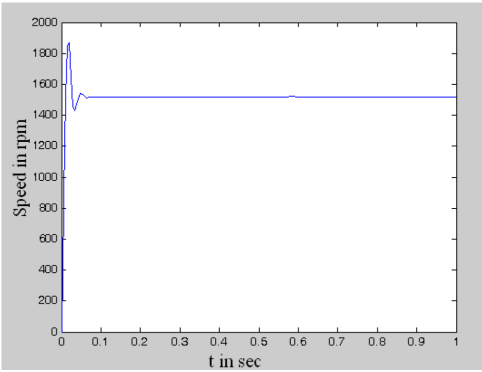


Figure 4 Simulation results for compensated system with P-controller

The simulation result shows an overshoot of 26.67%, settling time 0.05sec settling time, and a steady state error of 5%.

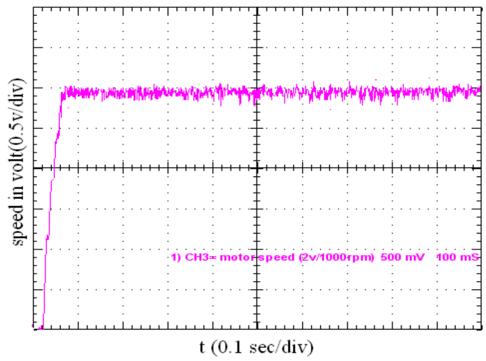


Figure 5 Practical results for compensated system with P-controller

The practical system presents a minimum overshoot due to the proposed saturation limit for the value of the controller output to avoid the armature current rise from damaging the coils in the armature. Figure 6 presents a good agreement between theoretical and experimental results.

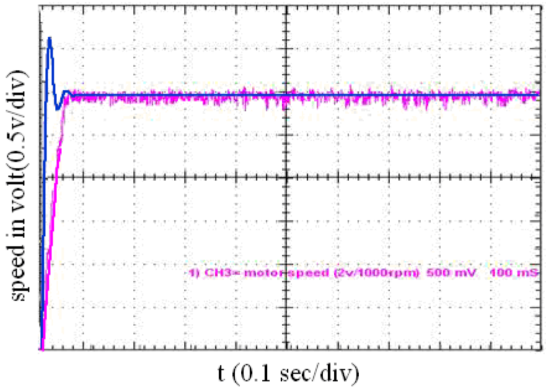


Figure 6 Response with P-controller for both simulated and actual systems

For the compensated system with PD-controller the simulation result is shown in figure 7, and the practical result is shown in figure 8.

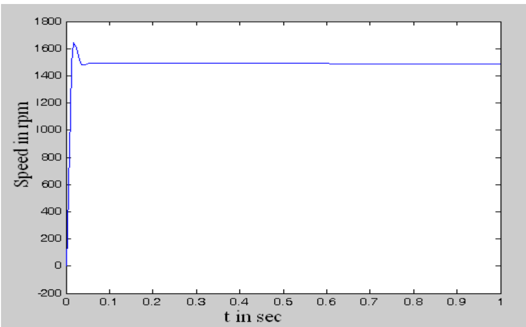


Figure 7 Simulation results for compensated system with PD-controller

The simulation result shows an overshoot of 10%, settling time 0.03sec settling time, and a steady state error of 3%.

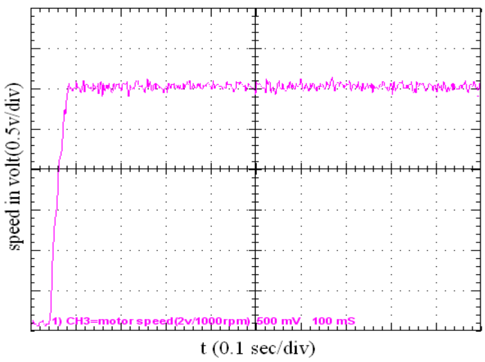


Figure 8 Practical results for compensated system with PD-controller

The practical system presents a minimum overshoot due to the proposed saturation limit. Figure 9 presents a good agreement between theoretical and experimental results.

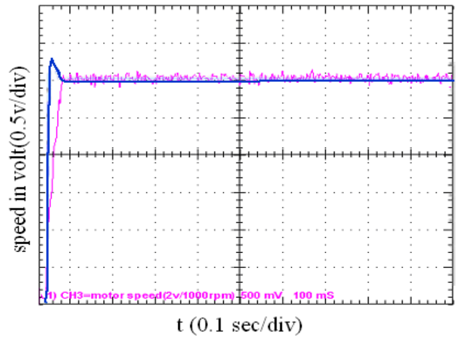


Figure 9 Response with PD-controller for both simulated and actual systems

For the compensated system with PI-controller the simulation result is shown in figure 10, and the practical result is shown in figure 11.

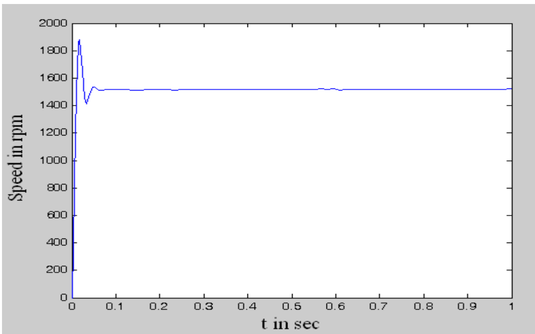


Figure 10 Simulation results for compensated system with PI-controller

The simulation result shows an overshoot of 26.7%, settling time 0.055sec settling time, and no steady state error.

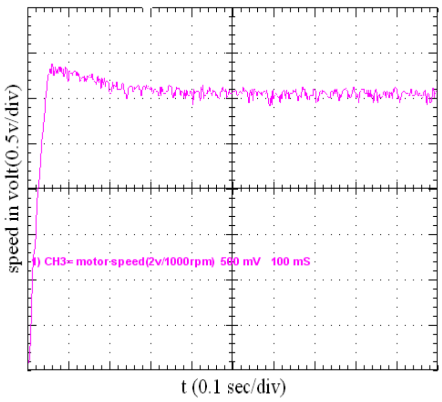


Figure 11 Practical results for compensated system with PI-controller

The practical system presents an overshoot of 11.67%. Figure 12 presents a good agreement between theoretical and experimental results.

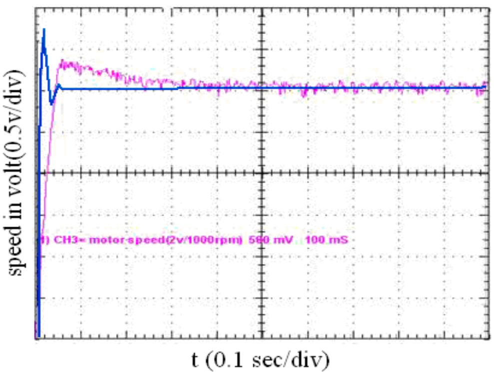


Figure 12 Response with PI-controller for both simulated and actual systems

For the compensated system with PID-controller the simulation result is shown in figure 13, and the practical result is shown in figure 14.

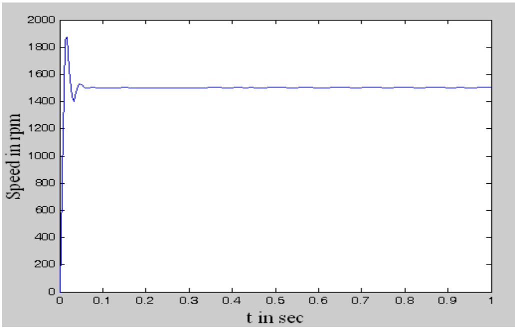


Figure 13 Response with PID-controller for both simulated and actual systems

The simulation result shows an overshoot of 23.3%, settling time 0.05sec settling time, and no steady state error.

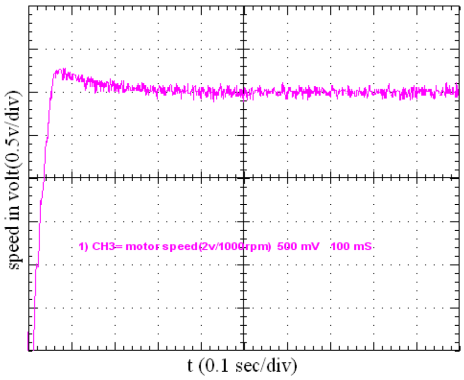


Figure 14 Response with PID-controller for both simulated and actual systems

The practical system presents an overshoot of 6.67%. Figure 15 presents a good agreement between theoretical and experimental results.

2.5 Design of Artificial Neural Network Controller

To design an artificial neural network controller with the required accuracy and speed of adaptation a large number of training data is required; this training data can be achieved using the inverse model control method shown below in figure 16.

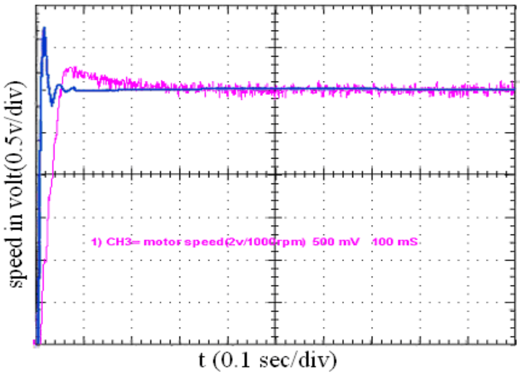


Figure 15 Response with PID-controller for both simulated and actual systems

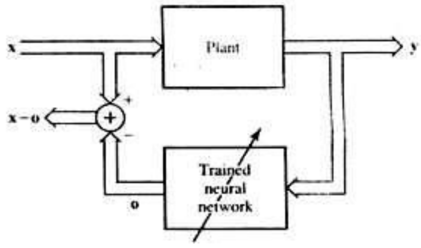


Figure 16 Inverse model control method

Figure 17 presents the proposed neural controller which consists of three inputs and single output this number of input/output can be achieved from equation 5, which describes the dc motor discrete model.

Figure 17 the proposed ANN controller

The training data can be achieved by using the model shown in figure 16 by applying an input signal **x** to the plant model to obtain the plant output which is considered as the input data to the proposed neural network and the value of the actual input to the plant is considered the desired output of the trained neural network

In this paper the selected input to the neural network is

$$\omega(k+1) = 50\sin(2\pi kT/7) + 45(2\pi kT/3) \quad (4)$$

And the desired response is obtained after rearranging equation 3 as follows

$$v(k+1)=0.08203\omega(k+2)-0.1513\omega(k+1)+0.0694\omega(k) \quad (5)$$

Several feedforward ANN models were designed and tested in this paper. These are combination of one learning algorithm, two transfer functions and many different structures selected among others due to their best generalizing ability in comparison with the all other tried combinations. The used learning algorithms were the Levenberg-Marquardt, while the transfer functions were the logarithmic sigmoid and the pure-line. Table (1) shows the ANN arrangements proposed in this paper.

The ANN can be now trained according to the block diagram shown in figure 16 by using MATLAB simulation. Sum of the Squared Error (SSE) graph shown in figure 18 which shows the progress of the neural network during the training process also the error between the actual and target is shown in figure 19.

Table (1) ANN controller arrangements (T=1ms)

Layers	Layer 1	Layer 2	Layer 3
No. of neurons	8	6	2
Transfer function	Logistic sigmoid	Pure line	Pure line
No. of training data	30,000		
Error goal	0.00015		
Learning rate	0.04		
Momentum	0.05		
Learning algorithm	Levenberg-Marquardt		

The response of the target output is shown in figure 20 and the graph is drawn against the number of epochs, the actual response of the ANN is shown in figure 21.

Also the actual and target data has been drawn on the same figure 22; from this figure it can be proved that a good agreement between target and actual data has been achieved.

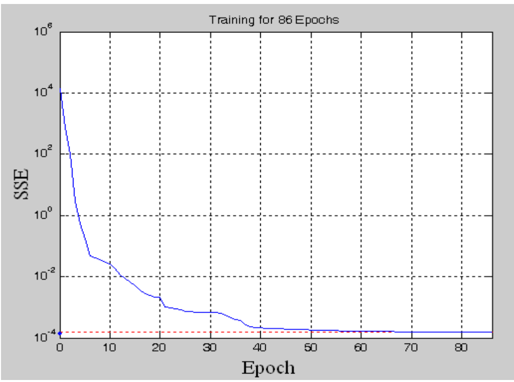


Figure 18 the sum squared error (SSE)

Figure 23 shows the proposed ANN controller after fitting it to the dc motor reference model. Figure 24 shows the response of the actual ANN to a desired speed of (W_d =150 rad/s), which is shown by the dashed line in figure 24, the actual speed

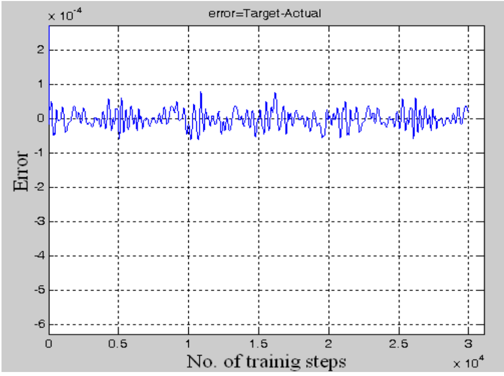


Figure 19 the error (target-actual)

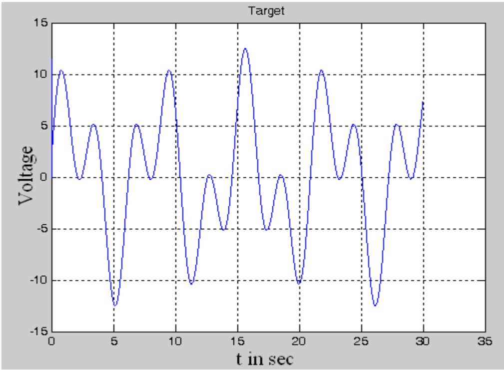


Figure 20 the target signal

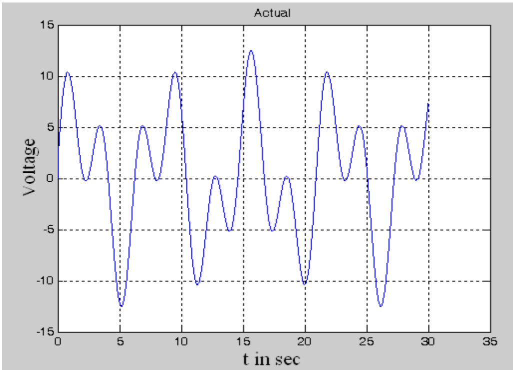


Figure 21 the actual signal

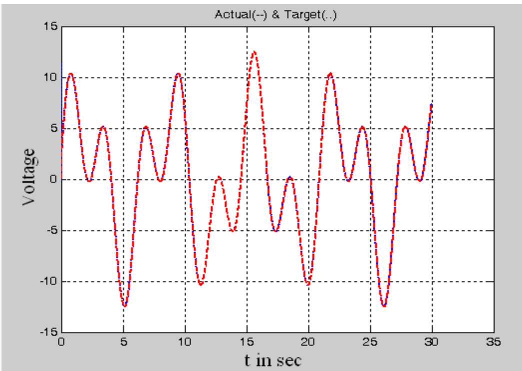


Figure 22 target signal, and actual

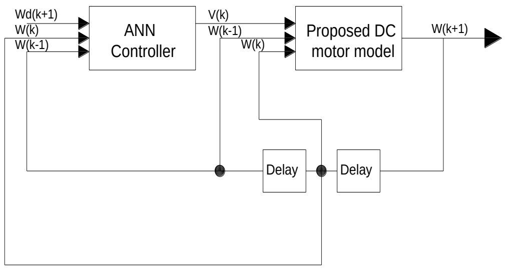


Figure 23 the proposed ANN controller configuration

reaches the desired speed in a settling time of 0.18sec (for ±5% of the desired output), with an over shoot of 20%, and zero steady state error.

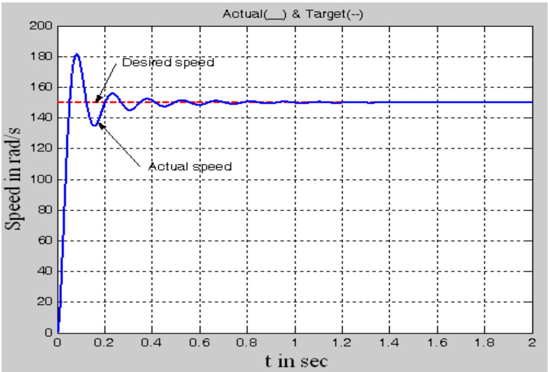


Figure 24 Overall system response

Comparison between the achieved response obtained from the inverse model ANN controller figure 24 and the response achieved with the traditional methods is presented in table 2.

Table (2) Comparison between conventional and intelligent controllers

	Open loop	**P**	**PD**	**PI**	**PID**	**ANN**
Settling time(s)	0.4	0.04	0.03	0.055	0.05	0.18
Over shoot(%)	-	25.3	10	26.7	23.3	20
Rise time(s)	-	0.02	0.02	0.02	0.02	0.025
e_{ss} (%) for 50% load	large	5	3	0	0	0

CONCLUSION

1. In this paper a comparative study is made between conventional controllers and artificial neural network controller.
2. A practical dc motor is used in simulation results. Experimental results are compared to show the good agreements between theoretical and experimental results.
3. ANN controller proposed in this paper has been tested with the proposed dc motor model to follow any arbitrarily selected speed.

REFERENCES

[1] Omar M.A. Salim, 'Analysis and Design of Control Systems Using Artificial Intelligence', M.Sc Thesis, Banha University 2006.

[2] Ying Wu, Zhiqiang Lv, Dabin Hu, Lin He, Yingyun Huang, Shijian Zhu, and Dingfang Chen, 'The BP Neural Network Intelligent Evaluation System Of Raft Vibration-isolating Unit Design', Proceeding of the 2002 International conference on control and automation, Xiamen, China, June 2002.

[3] Siri Weerasooria, and M. A. El-Sharkawi 'Identification And Control Of DC Motor Using Back Propagation Neural Networks', IEEE Trans. On Energy Conversion, vol. 6, No. 4, December 1991.

[4] Carmadi Machbub, Ary Setijadi Prihatmanto, Yoseph Dwi Cahaya: 'Design And Implementation Of Adaptive Neural Networks Algorithms For DC Motor Speed Control System Using Simple Microcontroller', IEEE PEDS 2001- Indonesia.

[5] M. D. Minkova, D. Minkov *, J. L. Rodgerson, R.G. Harly , 'Adaptive neural speed controller of a DC motor', ELSEVIER, Electric Power Systems Research 47(1998) 123-132.

[6] Hassanzadah, S. Khanmohammadi, M.B.B. Sharifian, and J. Jiang, 'A SISO Discrete Control System Containing Neural Estimator And Neural Controller', IEEE 2000.