Military Technical College
Kobry El-Kobba,
Cairo, Egypt

11-th International Conference
on Aerospace Sciences &
Aviation Technology

Adaptive Control of Brushless DC Motor Using Neural Network Identification and Pole Shifting Controller

A. Eliwa* W. Sabry* A. El-Wakeel* U. Abou-Zayed*

ABSTRACT

In this paper adaptive control of a brushless DC motor (BLDCM) using neural network identification and pole shifting (PS) controller is presented. Proper system identification is one of the important factors that gives a good controller performance. This means that when the model parameter estimates are good, the controller output is good, whereas if the model parameter estimates are bad then almost surely the computed control will be bad. Proper selection of the identified system model order is also investigated. A comparison study between fuzzy logic controller and the proposed controller is also investigated.

* Egyptian Armed Forces

1- Introduction

An automatically adapting controller becomes a very attractive proposition where a real plant is changing with respect to time and little is known about the plant. Self-tuning control is one form of adaptive control. Essentially, the self-tuning control algorithm consists of two stages: first a fairly simple model of the plant (generally a third order model in the case of a BLDCM application) is identified. The parameters of the model are updated regularly in order to make the model of the plant. Second, the updated parameters of the plant model are used in the controller to obtain an appropriate feedback control signal based on the assumption that the updated plant model parameters define the plant. In this paper, adaptive neural network identification is investigated.

Neural networks are a special kind of network that can learn from examples. This involves adjusting the weights that define the strength of connection between the neurons in the network. This can be interpreted as a system identification problem [1-4] with the advantage that many of the ideas and results in estimation theory can be applied to provide insight into the neural network problem irrespective of the specific application.

System identification and control using neural networks was proposed in an archival contribution in the 1990 [2]. It was demonstrated that neural networks can be used effectively for the identification and control of non-linear dynamical systems.

The problem formulation is described next without reference to any particular network. This is followed by a discussion of the architectures and learning algorithms in relevance to the BLDCM identification. In this paper adaptive linear neural network (ADALINE) will be used. The ADALINE network first proposed by Widrow [5] consisted of a non-linear limiting function in the output. To formulate the linear regression problem as an ANN, a linear transfer function has been used instead of the limiting function in the output of the ADALINE.

2- System Model

Fig.(1) shows the structure of the adaptive control system connected to a brushless DC motor drive. The adaptive control system consists of neural network identification and a pole shifting controller. In the proposed adaptive controller, the brushless DC motor is identified by a third order discrete model of the form:

$$A(z^{-1})y(t) = B(z^{-1})u(t) + \zeta(t) \tag{1}$$

where $A(z^{-1})$ and $B(z^{-1})$ are polynomials in the backward shift operator z^{-1} and are defined as:

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} \tag{2}$$

$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} \tag{3}$$

and the variables y(t), u(t) and ζ(t) are the system output, system input and white noise respectively.

The continual on-line model parameters, a_i and b_i, is called recursive parameter estimation such that, at the commencement of each sample interval, the estimations obtained during the previous recursion are made available and form a startup point ready to be updated. Many different techniques exist for updating these parameter estimates.

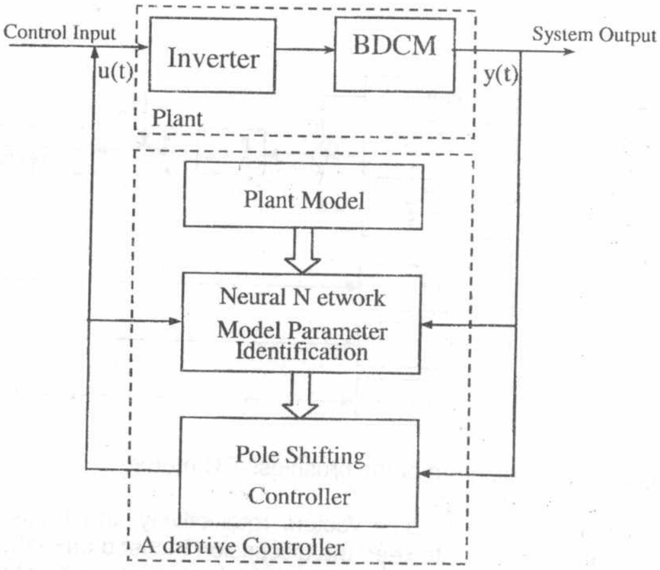

Fig.(1) Block Diagram of a Self-Tuning Controller

3- Modeling of Brushless DC Servo Motor

The brushless DC servo motor considered in the paper is a three-phase permanent magnet synchronous motor with sinsoidally back EMF. The stator windings are identical, displaced by 120° and sinusoidal distributed [6]. The block diagram of the brushless dc motor is shown in Fig.(2).

4- Problem Formulation

Consider the non-linear relationship:

$$y(t) = f\big(y(t-1), \;\ldots\; y(t-n_y),\; u_i(t-1), \;\ldots\; u_i(t-n_u)\big) + \zeta(t) \qquad (4)$$

where

$$y(t) = [y_1(t), \ldots, y_m(t)]^T, \qquad (5)$$

$$u_i(t) = [u_1(t), \ldots, u_r(t)]^T, \qquad (6)$$

$$\zeta(t) = [\zeta_1(t), \ldots, \zeta_m(t)]^T \qquad (7)$$

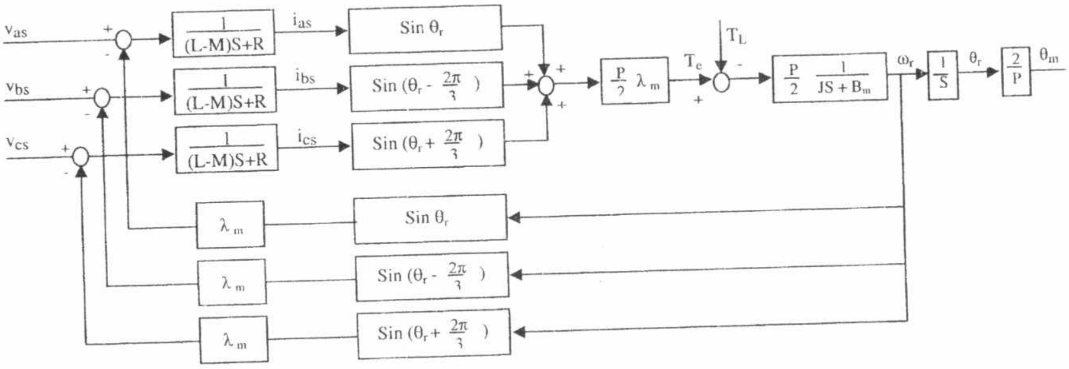

Fig.(2) Block diagram of the brushless DC motor

are the system output, input and noise vectors respectively and **f**(.) is some vector valued nonlinear function. In order to keep the discussion focused on neural networks in this section, the analysis will relate to the model of Eqn. (4) with the aim of approximating the underlying dynamics **f**(.) using neural networks. Throughout the paper, the network input/output relationship will be defined as:

$$\hat{y}(t) = f(V(t)) \qquad (8)$$

where *V(t)* is the neural network input vector, and the model predicted output is defined as:

$$\hat{y}(t) = f\big(y(t-1), \;\ldots\; y(t-n_y),\; u_i(t-1), \;\ldots\; u_i(t-n_u)\big) \qquad (9)$$

5- ADALINE Architecture and Learning Algorithm

The ADALINE structure is shown in Fig.(3). The output of the ADALINE can be expressed as a linear combination of the input signal.

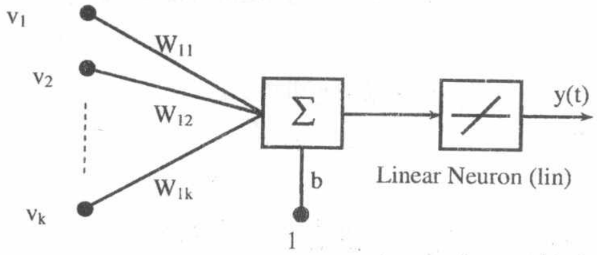

Fig.(3) ADALINE Network

$$\hat{y} = \boldsymbol{WV} + b \tag{10}$$

where ***V*** is the ADALINE input vector, ***W*** is the weight vector and *b* is the bias.

The neuron uses a linear transfer function which simply outputs the value passed to it. This can be trained to learn *affine* (allocating finite values to finite quantities) function of the inputs, or to find a linear approximation to a non-linear function.

<u>Widrow-Hoff learning</u>

The Widrow-Hoff learning algorithm [5] adjusts the weights and bias, b, of the ADALINE so as to minimize the mean-squared error (MSE) given by Eqn. (11):

$$MSE = \frac{1}{N}\sum_{n=1}^{N} e^2(n) = \frac{1}{N}\sum_{n=1}^{N}(y(t) - \hat{y}(t))^2 \tag{11}$$

The MSE performance index for the ADALINE network is a quadratic function. Thus, the performance index will either have one global minimum, a weak minimum or no-minimum. Specifically, the characteristics of the input vectors determine whether or not a unique solution exists.

The learning algorithm is obtained using the following equations:

$$\frac{\partial e^2(n)}{\partial w_{1j}} = 2e(n)\frac{\partial e(n)}{\partial w_{1j}} \tag{12}$$

and

$$\frac{\partial e^2(n)}{\partial b} = 2e(n)\frac{\partial e(n)}{\partial b} \tag{13}$$

Next looking at the partial derivatives with respect to the weights:

$$\frac{\partial e(n)}{\partial w_{1j}} = \frac{\partial\left[y(t) - \hat{y}(t)\right]}{\partial w_{1j}} = \frac{\partial\left[y(t) - (\mathbf{WV}(n) + b)\right]}{\partial w_{1j}} \tag{14}$$

$$\frac{\partial e(n)}{\partial w_{1j}} = \frac{\partial\left[y(t) - \left(\sum_{i=1}^{k} w_{1i}v_i(n) + b\right)\right]}{\partial w_{1j}} \tag{15}$$

Here $v_i(n)$ is the i^th^ element in the input vector $\boldsymbol{V}(n)$ in the n^{th} iteration. From Eqn. (14) the following equations follow:

$$\frac{\partial e(n)}{\partial w_{1j}} = -v_j(n)$$

and

$$\frac{\partial e(n)}{\partial b} = -1 \tag{16}$$

These results form the learning rule and can be written in the matrix form as:

$$\begin{aligned} \mathbf{W}(n+1) &= \mathbf{W}(n) + 2\eta\, e(n)\, \mathbf{V}^T(n) \\ b(n+1) &= b(n) + 2\eta\, e(n) \end{aligned} \tag{17}$$

where η, the learning factor, decides the speed of convergence of the iterative procedure. If η is large, learning occurs quickly, but if η is too large it leads to instability and the errors may even increase. To ensure stable learning, the learning rate must be less than the reciprocal of the largest eigenvector of the correlation matrix [7].

6- ADAPTIVE POLE-SHIFTING CONTROLLER

Consider the system shown in Fig. (3) is modeled by:

$$A(z^{-1})y(t) = B(z^{-1})u_i(t) + \zeta(t) \tag{18}$$

where *y(t), $u_i(t)$, ξ(t)* are system output, system input and white noise respectively. *$A(z^{-1})$, $B(z^{-1})$* take the form:

$$A(z^{-1}) = 1 + a_1 z^{-1} + ... + a_i z^{-i} + ... + a_{n_a} z^{-n_a} \qquad (19)$$

$$B(z^{-1}) = b_1 z^{-1} + ... + b_i z^{-i} + ... + b_{n_b} z^{-n_b} \qquad (20)$$

where $n_a \geq n_b + l$.

The system parameters a_i, b_i are fixed or obtained on-line using a system parameter identifier. The computation of the control signal *u(t)* using the pole-shifting (PS) algorithm[8, 9] is briefly described below. Assume the feedback loop has the form:

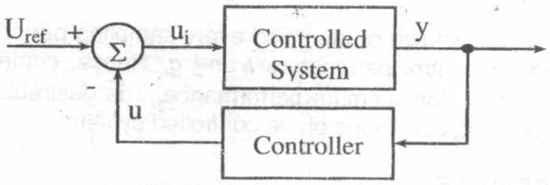

Fig.(3) Block diagram of the controller

$$\frac{u(t)}{y(t)} = -\frac{G(z^{-1})}{F(z^{-1})} \qquad (21)$$

Where

$$F(z^{-1}) = 1 + f_1 z^{-1} + ... + f_i z^{-i} + ... + f_{n_f} z^{-n_f} \qquad (22)$$

$$G(z^{-1}) = g_o + g_1 z^{-1} + ... + g_i z^{-i} + ... + g_{n_g} z^{-n_g} \qquad (23)$$

and

$$n_f = n_b - 1, \qquad n_g = n_a - 1 \qquad (24)$$

From Eqns. (21 to 24) the closed-loop characteristics polynomial *$T(z^{-1})$* can be derived as:

$$T(z^{-1}) = A(z^{-1})F(z^{-1}) + B(z^{-1})G(z^{-1}) \qquad (25)$$

The PS control algorithm makes *$T(z^{-1})$* take the form of *$A(z^{-1})$* but the pole locations are shifted by a factor α, i.e.

$$A(z^{-1})F(z^{-1}) + B(z^{-1})G(z^{-1}) = A(\alpha z^{-1}) \qquad (26)$$

Expanding both sides of Eqn. (26) and comparing the coefficients gives:

$$\begin{bmatrix} 1 & 0 & . & 0 & b_1 & 0 & . & 0 \\ a_1 & 1 & . & 0 & b_2 & b_1 & . & 0 \\ . & a_1 & . & . & . & b_2 & . & . \\ a_{n_a} & . & . & 1 & b_{n_b} & . & . & b_1 \\ 0 & a_{n_a} & . & a_1 & 0 & b_{n_b} & . & b_2 \\ . & 0 & . & . & . & 0 & . & . \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & . & a_{n_a} & 0 & 0 & . & b_{n_b} \end{bmatrix} \begin{bmatrix} f_1 \\ . \\ \vdots \\ f_{n_f} \\ g_0 \\ . \\ \vdots \\ g_{n_g} \end{bmatrix} = \begin{bmatrix} a_1(\alpha - 1) \\ a_2(\alpha^2 - 1) \\ . \\ \vdots \\ a_{n_a}(\alpha^{n_a} - 1) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad (27)$$

or in a matrix form:

$$\boldsymbol{M}.w(\alpha) = \boldsymbol{L}(\alpha) \qquad (28)$$

Parameters a_i and b_i are known or identified every sampling period. If α is known, Eqn. (28) can be solved for control parameters f_i and g_i. Hence, control signal *u(t)* can be obtained from Eqn. (18). For optimum performance, it is desirable to modify α on-line according to the operating conditions of the controlled system.

7- SIMULATION RESULTS

The effectiveness of the proposed adaptive control using neural network identification and poll shifting controller on the transient and dynamic characteristics of the rotor speed of the brushless dc motor is investigated. The speed control of the drive was also designed and simulated with fuzzy control, in order to compare the performance with the respective proposed controller. The BLDCM was also operated at full load for 0.3 second, then the load torque was decreased to 70% of its full load value for a period of 0.4 second and the load torque return to its full load value for a period of 0.3 second. The simulation results are shown in Fig.(4) and Fig.(7). From the above Figures, we find that the speed overshoot with adaptive controller is smaller than that with fuzzy controller. Also the recovery time with adaptive controller is shorter than that with fuzzy controller.

References

[1] S. Chen and S.A. Billings, Neural Networks for Nonlinear Dynamic System Modeling and Identification, Advances in Intelligent Control, Taylor and Francis, London, 1990.

[2] K.S. Narendra and K. Parthasarthy, "Identification and Control of Dynamic Systems Using Neural Networks", IEEE Trans. on Neural Networks, vol. 1, no. 1, March 1990, pp. 4-27.

[3] S. Chen, S.A. Billings and P.M. Grant, "Non-linear Systems Identification Using Neural Networks", Int. J. Control, vol. 51, 1990, pp. 1191-1214.

[4] S. Chen, S.A. Billings and P.M. Grant, "Recursive Hybrid Algorithm for Non-Linear System Identification Using Radial Basis Function Networks", Int. J. Control, vol. 55, no. 5, 1992, pp. 1051-1070.

[5] B. Widrow, Generalization and Information Storage in Network of ADALINE Neuron, Self-Organizing Systems, M. Yovitz, G. Jacobi, and G. Goldstein (Eds.), Spartan Books, Washington DC, 1962, pp. 435-461.

[6] T. L. Chen and Y. C. Wa, " Design of brushless DC position servo systems using integral variable structure approach" IEE proceedings-B, vol. 140, no.1 January 1993.

[7] M. T. Hagan, H.B. Demuth and M.H. Beale, Neural Network Design, PWS Publishing, Boston, 1996.

[8] G.P. Chen, "An Adaptive Self-Optimizing Power System Stabilizer", Ph.D. Dissertation, The University of Calgary, Calgary, Canada, February 1994.

[9] Ramakrishna Gokaraju, "Identification and Control Tools", Ph.D. Dissertation, The University of Calgary, Calgary, Canada, May 2000.

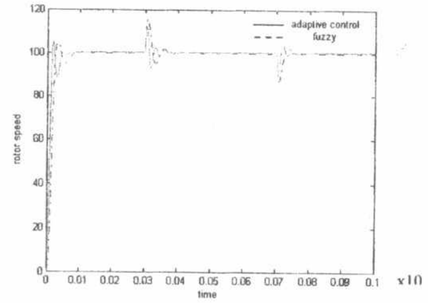

Fig.(4) The speed response of the adaptive controller & fuzzy controller at 0.7 full load torque

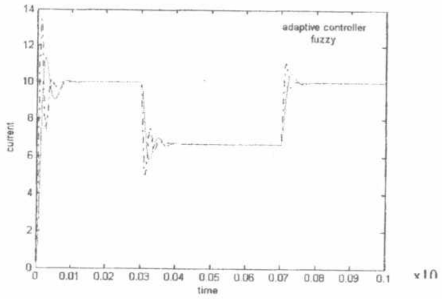

Fig.(5) current response with adaptive controller & controller at 0.7 full load torque fuzzy controller

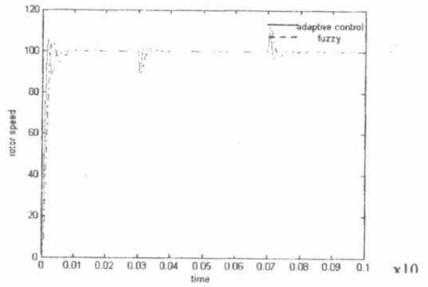

Fig.(6) The speed response of the adaptive controller & fuzzy controller at 1.3 full load torque

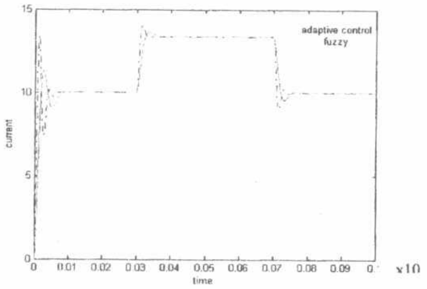

Fig.(7) current response with adaptive controller & controller at 1.3 full load torque fuzzy controller