NEW APPROXIMATIONS WITH DIFFERENT TYPE NEURAL NETWORK STRUCTURES AND ALGORITHMS FOR PERFORMANCE INCREMENT OF AC DRIVE CONTROL SYSTEMS

Kayhan GÜLEZ

Yıldız Technical University, Electrical-Electronics Faculty, Electrical Engineering Department, Yıldız-İSTANBUL

Geliş Tarihi: 10.01.2003

AC SÜRME DEVRELERİNE SAHİP KONTROL SİSTEMLERİNDE PERFORMANS ARTTIRIMI İÇİN FARKLI TİP SİNİR AĞI YAPILARI VE ALGORİTMALARI İLE YENİ YAKLAŞIMLAR

ÖZET

Bu makalede, DSP tabanlı bir kontrol biriminin, AC sürme devresine sahip motor kontrol sistemlerine performans arttırımı icin uygulanmasinda farkli tip sinir agları ile yaklaşımlar temel ve önemli bir çalışma olarak dikkate alınmıştır. Burada Fast Back-propagation algoritması motor kontrol uygulamalarına ve AC sürme deverelerine ilk kez uygulanmaktadır. Algoritma, sistem hatasını azaltmada ve çok yüksek işleme hızını sağlamada eş zamanlı olarak oldukça başarılı sonuçlar vermektedir.

ABSTRACT

In this paper, the new approximations with different type neural networks for the application of AC drive motor control systems to increase the performance using a DSP based control according to variable load conditions, as a basic and important study is considered. Fast Back-propagation algorithm is firstly used for motor control applications and AC Drives through the paper. The algorithm gives considerably important success to decrease the system error and ensure very high data processing time simultaneously.

1. INTRODUCTION

In recent years, control systems have assumed an increasingly important in the development and advancement of modern civilization and technology. One of the major problems of motor drives is that they are traditionally designed with relatively inexpensive analogue components. The weaknesses of analogue systems are their susceptibility to temperature variations and the component aging. Another difficulty is upgrading the system for new needs [1]-[8]. The usage of changeable load conditions affects the performance of AC drives-especially induction motor control systems directly. In recent years, all new type control methods are being designed to improve digital control structures. Digital control structures eliminates drifts, solve complex mathematical equations and by using programmable processor it can be easily upgraded for new control algorithms. Digital Signal Processors (DSP) go further; their high performance allows them to perform high resolution control and minimise control loop delays. By the help of their fast processor which works parallel, many complicated control algorithms can be implemented easily [4], [6], [8]. With the importance of digital control structures, it was used a TMS320C50 processor board to implement the dynamic speed controller to induction motor. New technologies such as neural networks, fuzzy logic etc., give the opportunity to upgrade, update and improve the system according to new needs easily and accurately.

The basic ingredients of a control system can be described by objectives of control system components and results or outputs. In block diagram form, the basic relationship between these components is illustrated in Figure 1 [8].

Almost, every control system includes a servo motor with a soft feed back. They need good results to implement to system. Thus, it is designed a dynamic speed controller (including PI structure) to use the main advantages of robust structure of induction motor under the control of a DSP-based system.

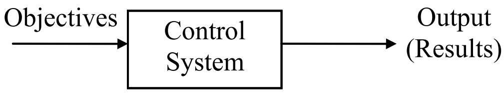


Figure 1. Basic components of a control system

Field Oriented Control (FOC) method including PI parameter adaptation is used for the control structure of DSP implementation. FOC is based on three major points: the machine current and voltage space vectors, the transformation of a three phase speed and time dependent system into a two co-ordinate time invariant system and effective Pulse With Modulation (PWM) pattern generation. Thanks to these factors, the control of AC machines acquires every advantage of DC machine control and frees itself from the mechanical commutation drawbacks. Furthermore, this control structure, by achieving a very accurate steady state and transient control, leads to high dynamic performance in terms of response times and power conversion.

FOC consists of controlling the stator currents represented by a vector [3]. This control is based on projections which transform a three-phase time and speed dependent system into a two co-ordinate (d and q co-ordinates) time invariant system. These projections lead to a structure similar to that of a DC machine control. As FOC is simply based on projections the control accurate in every working operation (steady state and transient) and independent of the limited bandwidth mathematical model. The FOC thus solves the classic scheme problems, in the following ways,

- the ease of reaching constant reference (torque component and flux component of the stator current)
- the ease of applying direct torque control because of the expression of the torque in the (d, q) reference frame.

By maintaining the amplitude of the rotor flux at a fixed value we have a linear relationship between torque and torque component. The torque can then be controlled by controlling the torque component of stator current vector by using Clark and Park transforms [15].

2. THE MATHEMATICAL STRUCTURE OF INDUCTION MOTOR AND DYNAMIC SPEED CONTROLLER

Following equations show the induction motor mathematical model and PI controller [10], [12], [15].

$$\vec{v}_s = R_s \vec{i}_s + L_s \frac{d\vec{i}_s}{dt} + L_m \frac{d}{dt}(\vec{i}_r e^{j\theta_r}) \tag{1}$$

$$0 = R_r \vec{i}_r + L_r \frac{d\vec{i}_r}{dt} + L_m \frac{d}{dt}(\vec{i}_s e^{-j\theta_r}) \tag{2}$$

$$J\frac{dw_r}{dt}+Bw_r=T_e-T_d$$
$$=\frac{P}{2}L_m I_m\left[\vec{i}_s(\vec{i}_r e^{j\theta_r})^*\right]-T_d \qquad \text{* represents a matrix vector.} \tag{3}$$

$$\frac{d\theta_r}{dt}=w_r \tag{4}$$

Thus, the torque equation can be shown as,

$$T_e=\frac{P}{2}\frac{L_m}{L_r}(i_{sq}\psi_{rd}-i_{sd}\psi_{rq}). \tag{5}$$

After some mathematical computations, the following equation is obtained.

$$T_e=\frac{P}{2}\frac{L_m}{L_r}\psi_r i_{sq} \tag{6}$$

$$T_e^*=K_t i_{sq}^* \tag{7}$$

$$\left.\frac{w_m(s)}{w_m^*(s)}\right|_{T_d=0}=\frac{K_iK_t}{Js^2+(B+K_pK_t)s+K_iK_t} \tag{8}$$

comparing with a second order system, damping factor and natural frequency,

$$\xi=\frac{B+K_pK_t}{2(JK_iK_t)^{1/2}}; \quad w_d=\left(\frac{K_iK_t}{J}\right)^{1/2} \tag{9}$$

and bandwidth is

$$BW=w_d\sqrt{1-2\xi^2+(2-4\xi^2+4\xi^4)^{1/2}} \tag{10}$$

related to response time for equation 8,

$$t_c\approx\frac{3.875}{w_d}\approx\frac{0.65\times3.875}{BW} \qquad \xi=1 \;\; for \tag{11}$$

$$K_i=Jw_d^2/K_t \quad ; \quad K_p=(2Jw_d-B)/K_t \tag{12}$$

The PI controller block diagram is depicted in Figure 2. Table 1 shows the parameters of the tested motor.

3. ARTIFICIAL NEURAL NETWORKS (ANN)

Artificial Neural Networks (ANN) are successfully used in a lot of areas such as control, early detection of electric machine faults, digital signal processing in our daily technology [12]-[13]. The feed-forward neural network is usually trained by a back-propagation training algorithm first proposed by Rumelhart, Hinton, and Williams in 1986. This was the effective usage of it only after 1980s. With the help of speedy computers, NNs are more realistic and easily updateable today. The distributed weights in the network contribute to the distributed intelligence or associative memory property of the network. With the network initially untrained, i.e., with the weights selected at random, the output signal pattern will totally mismatch the desired output

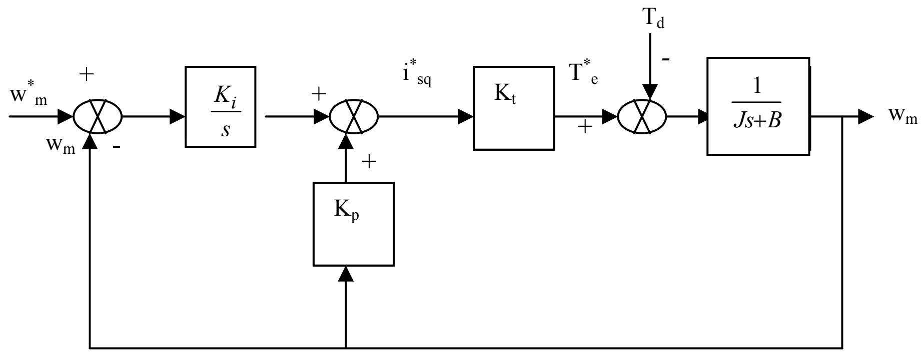


Figure 2. The block diagram of PI controller

Table 1. The parameters of the tested motor

Power rating	0.56 kW
Rated voltage	380 V
Pole pairs	2
Total inertia (J)	2.33 g-m^2
R_s	0.17 Ω
R_r	0.33 Ω
L_s	31.4 mH
L_r	33.4 mH
L_m	3.18 mH

pattern for a given input pattern. The actual output pattern is compared with the desired output pattern and the weights are adjusted by the supervised back-propagation training algorithm until the pattern matching occurs, i.e., the pattern errors become acceptably small.

3.1. Classic Back-propagation Algorithm

Following equations show the basic ones of classic error back-propagation algorithm [12],[13].

$$o_j = f(net_j) = f(x) \;\; ise \;\; net_j = \sum_j^i w_{ji} o_i + \theta_j \tag{13}$$

$$E_p = \frac{1}{2} \sum_{j \in output} (t_{pj} - o_{pj}) \tag{14}$$

$$\delta_{pj} = (t_{pj} - o_{pj})$$

$$\Delta_p w_{ji} = -\varepsilon(\frac{\partial E_p}{\partial w_{ji}}) \tag{15}$$

$$\Delta_p \theta_j = -\varepsilon(\frac{\partial E_p}{\partial \theta_j})$$

If it is used "sigmoid" function, as the transfer (threshold) one in the operation element;

$$o_{pj} = \frac{1}{\sum_i 1 + e^{-w_{ji} o_{pi} + \theta_j}} \tag{16}$$

$$(\mathrm{netp}_j) = \sum_i \mathrm{w}_{ji} o_{pi} + \theta_j$$

it is derived the equation 16 and done necessary shortening;

$$\frac{\partial o_{pj}}{\partial netp_j} = o_{pj}(1 - o_{pj}) \tag{17}$$

If this is replaced in equation 16, for output element;

$$\delta_{pj} = (t_{pj} - o_{pj}) o_{pj} (1 - o_{pj}) \tag{18}$$

for hidden layer element;

$$\delta_{pj} = o_{pj}(1 - o_{pj}) \sum_k \delta_{pk} w_{kj} \tag{19}$$

If it is added (α) momentum term to the general equation set to speed up the computation of the algorithm, in the most general condition, it is given output and hidden layer equations as follows:

$$\begin{aligned} \Delta_p w_{ji}(t+1) &= \varepsilon \delta_{pj} o_{pi} + \alpha \Delta_p w_{ji}(t) \\ \Delta_p \theta_j (t+1) &= \varepsilon \delta_{pj} + \alpha \Delta_p \theta_j (t) \end{aligned} \tag{20}$$

Here; t: the number of learning cycles.
(α): a small positive number.

3.2. Fast Back-propagation Algorithm

The fast version [14], of the back-propagation algorithm is derived by sequentially minimizing the objective function G_k (λ), defined by equation 21, for k=1,2,.....m. The update equation for the synaptic weights w_{pq} is obtained as equation 22.

$$G_k(\lambda) = \lambda \sum_{i=1}^{n_0} \phi_2(e_{i,k}) + (1-\lambda) \sum_{i=1}^{n_0} \phi_1(e_{i,k}) \quad \forall\, k = 1,2,......, m \tag{21}$$

$$w_{p,k} = w_{p,k-1} + \alpha \varepsilon^0_{p,k}(\lambda) \hat{h}_k \tag{22}$$

If the output of the network is analog,

$$\varepsilon^{0}_{p,k}(\lambda) = \lambda(y_{p,k} - \hat{y}_{p,k}) + (1-\lambda)\tanh\left[\beta(y_{p,k} - \hat{y}_{p,k})\right] \quad (23)$$

On the other hand, if the network has binary outputs,

$$\varepsilon^{0}_{p,k}(\lambda) = (1 - \hat{y}^{2}_{p,k})(y_{p,k} - \lambda\,\hat{y}_{p,k}) \quad (24)$$

Because of its simplicity and fast, this algorithm provides an ideal basis for investigating the role of λ during training. Here, $\varepsilon_{p,k}$ and α are the output error and learning rate respectively. λ is given as,

$$\lambda=\exp(-\mu/E^2) \quad (25)$$

where μ is selected by user. μ is error decreasing sub-coefficient.

4. NEURAL NETWORK STRUCTURES FOR PERFORMANCE INCREMENT

4.1. Standard Order- Classic Back-propagation Algorithm Structure

For all the applications [9], [10], [11], It was used 0.55 kW, 2.6A, 220V, 50Hz, Cosϕ=0.79 3-phase squirral-cage induction motor. The processsor used in this work is, 40Mhz TMS320C50 DSP with 10k x 16 words of on-chip RAM which works parallel with TLC320C40 Analogue interface circuitry (AIC) with 14 bit. The processor is communicated with a PC through RS232 serial port [15]. The other motor coupled to the induction motor is a DC motor to work as generator to give the changeable load. The DC motor is 150-300V, 8.5-8.5A, 1-2 kW, 1500-3000 rpm, U_{Err}=220V, I_{Err}=0.6A including a digital tachometer. It is driven by a oto-transformer which has 220V/50 Hz AC input over a diode bridge, 385V/470μF capacitor and 2A circuit breaker. It was also used 2 Escor Pro5 5A Hall Effect sensors to sense the phase currents and 3 measurement circuits (mainly including LM741 operational amplificators) for comparative conditions of voltage levels especially for DSP inputs. DM7404 Hex Inverter was used for a time delay because of only one ADC-DAC unit. The signals were given over 74AHC573 Latch [15], and CD4016C to the control loop. Firstly, it was taken the numerical values belonging to the variables of the motor used in the training of neural network structure with the help of DSP based system. It was to say to run the control algorithm. The equations of the induction motor used in the application has been the tunable control algorithm of the DSP based system.

The language of ANN and Digital Signal Processor are C++ and Assembler respectively. In this application, for Classic Back-propagation Algorithm, as the input values to ANN, rotor speed (n) as angular speed (w), current (I) and power (P), as the output value to it torque (T) of the motor to increase the performance of the system are made and discriminated with success rate %0.0011 for 300000 iterations. There are 3 hidden layers including 5, 4, and 3 nodes in each one respectively. Table 2 and 3 show the full set of measurements result values of the system and test phase results respectively. It is depicted test phase results diagram in Figure 3.

4.2. Standard Order- Fast Back-propagation Algorithm Structure

In this application, for Fast Back-propagation Algorithm, as the input values to ANN, rotor speed (n) as angular speed (w), current (I) and power (P), as the output value to it torque (T) of the motor to increase the performance of the system are made and discriminated with success rate %0.0007 for 300000 iterations. Also, there are 3 hidden layers including 5, 4, and 3 nodes in each one respectively. Table 4 shows test phase results. It is depicted in Figure 4, test phase results comparative diagram.

Table 2. Some numerical values of the training set for all applications

w (rpm)	I (A)	P (%)	T (%)
0	0	0	0
10.4719	2.506	0.000571	0.025
64.9262	1.491	0.0406	0.1875
85.8701	1.432	0.0714	0.25
121.4749	0.835	0.1495	0.3625
142.4188	0.775	0.2213	0.4375
178.0235	0.651	0.327	0.5375
194.7787	0.537	0.389	0.5875
222.0058	0.412	0.4967	0.6625
289.0265	0.281	0.8542	0.875
301.5928	0.280	0.9334	0.9
286.9321	0.282	0.8653	0.9
247.1386	0.292	0.6217	0.7375
209.4395	0.416	0.4415	0.625
182.2123	0.598	0.339	0.55
136.1356	0.782	0.1836	0.4
69.1150	1.553	0.0469	0.2
14.6607	2.446	0.00241	0.05

Table 3. Test phase results for for Standard Order- Classic Back-propagation Algorithm Structure

w (rpm)	I (A)	P (%)	T (%)	ANN Results T (%)
43.9822	1.860	0.02053	0.125	0.132078
127.7581	0.831	0.165	0.3875	0.379243
207.3451	0.417	0.4456	0.625	0.627734
278.5545	0.283	0.792	0.8375	0.841526
194.7787	0.537	0.3895	0.5875	0.587541
100.5309	1.306	0.00946	0.3	0.292568
20.9439	1.847	0.00412	0.0625	0.060119

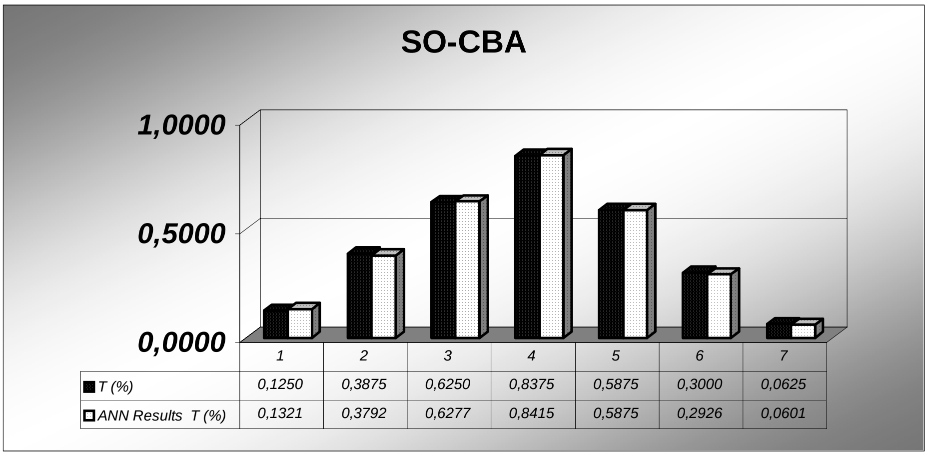


Figure 3. Test phase results comparative diagram with measurements for Standard Order-Classic Back-propagation Algorithm Structure

Table 4. Test phase results for Standard Order- Fast Back-propagation Algorithm Structure

w (rpm)	I (A)	P (%)	T (%)	ANN Results T (%)
43.9822	1.860	0.02053	0.125	0.127778
127.7581	0.831	0.165	0.3875	0.385945
207.3451	0.417	0.4456	0.625	0.626175
278.5545	0.283	0.792	0.8375	0.838156
194.7787	0.537	0.3895	0.5875	0.587261
100.5309	1.306	0.00946	0.3	0.299588
20.9439	1.847	0.00412	0.0625	0.061510

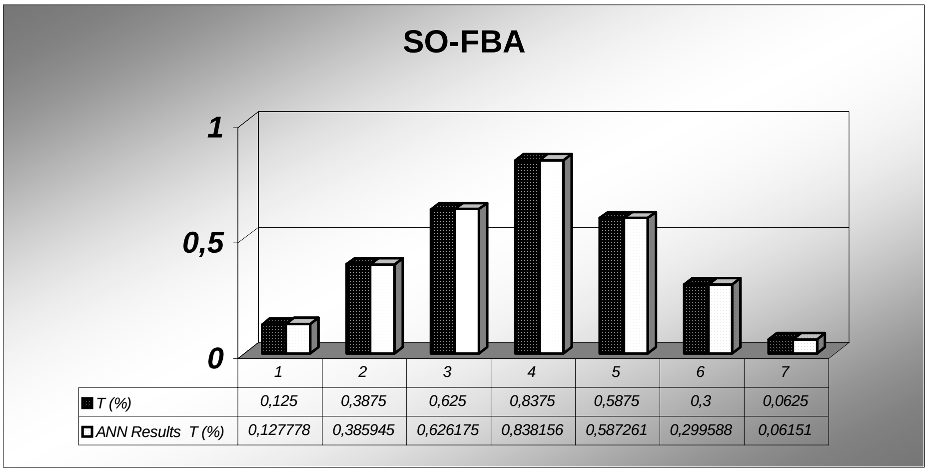


Figure 4. Test phase results comparative diagram with measurements for Standard Order- Fast Back-propagation Algorithm Structure

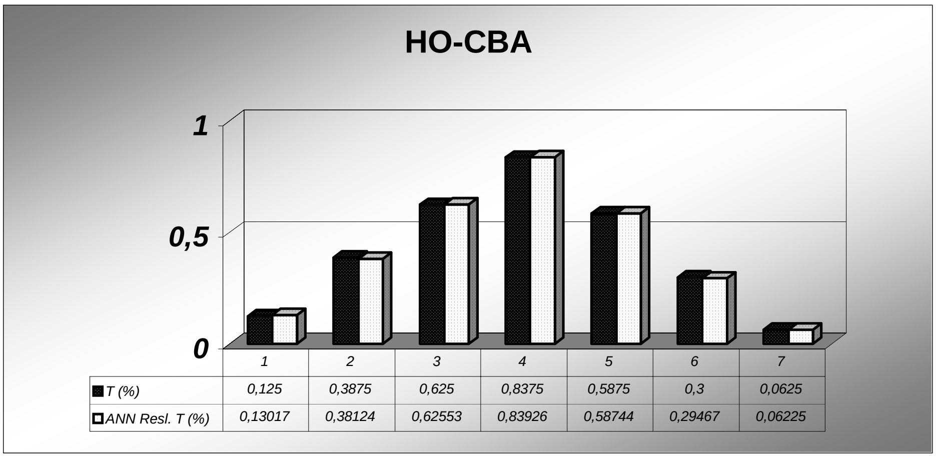


Figure 5. Test phase results comparative diagram with measurements for High Order- Classic Back-propagation Algorithm Structure

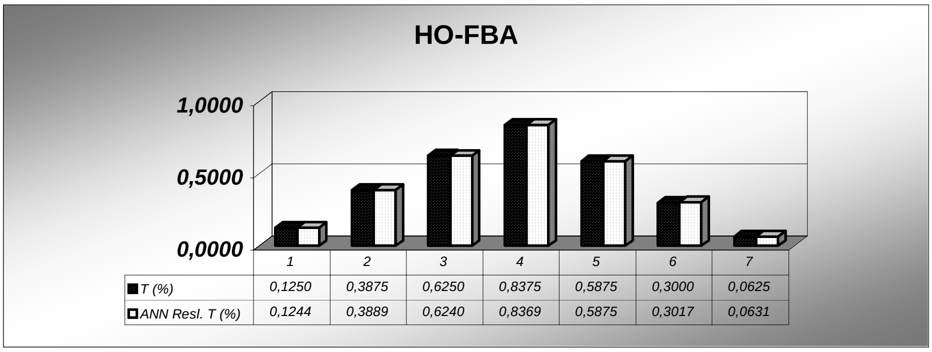
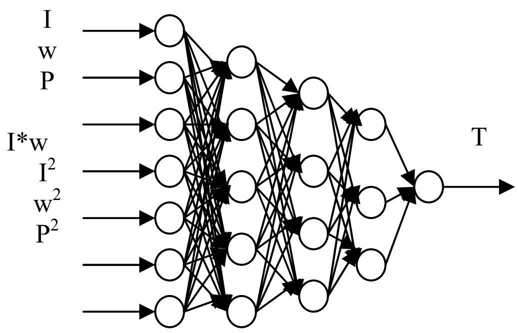


Figure 6. Test phase results comparative diagram with measurements for High Order- Fast Back-propagation Algorithm Structure



H.L.1-5, H.L.2-4, H.L.3-3

Figure 7. The ANN architecture of the system to increase the performance

4.3. High Order- Classic Back-propagation Algorithm Structure

In this application part, for Classic Back-propagation Algorithm, as the input values to ANN rotor speed (n) as angular speed (w), current (I), power (P), square of w, I, P and multiply of w and I, as the output value to it torque (T) of the motor with the same hidden layer structure such as previous ones to increase the performance of the system are made and discriminated with success rate, %0.0009 for 300000 iterations. Table 5 shows test phase results. It is depicted in Figure 5, test phase results comparative diagram.

4.4. High Order- Fast Back-propagation Algorithm Structure

In this application part, for Fast Back-propagation Algorithm, as the input values to ANN rotor speed (n) as angular speed (w), current (I), power (P), square of w, I, P and multiply of w and I, as the output value to it torque (T) of the motor with the same hidden layer structure such as previous ones to increase the performance of the system are made and discriminated with success rate, %0.0004 for 300000 iterations. Table 6 shows test phase results. It is depicted in Figure 6 and 7, test phase results comparative diagram and ANN structures for these all applications. Figure 8 and 9-12 show the whole control system and the different perspective photos of the experimental set.

Table 5. Test phase results for High Order- Classic Back-propagation Algorithm Structure

w (rpm)	I (A)	P (%)	w^2
43.9822	1.860	0.02053	1934.43
127.758	0.831	0.165	16322.1
207.345	0.417	0.4456	42991.9
278.554	0.283	0.792	77592.6
194.778	0.537	0.3895	37938.7
100.530	1.306	0.00946	10106.4
20.9439	1.847	0.00412	438.64

I^2	w*I	P^2	T (%)	• ANN Resl. T (%)
3.459	6692.3	0.00042	0.125	0.13017
0.690	11270.4	0.02722	0.3875	0.38124
0.173	7472.0	0.19855	0.625	0.62553
0.080	6213.6	0.62726	0.8375	0.83926
0.288	10937.7	0.15171	0.5875	0.58744
1.705	17237.5	0.00894	0.3	0.29467
3.411	1496.3	0.00001	0.0625	0.06225

Table 6. Test phase results for High Order- Fast Back-propagation Algorithm Structure

w (rpm)	I (A)	P (%)	w^2
43.9822	1.860	0.02053	1934.43
127.758	0.831	0.165	16322.1
207.345	0.417	0.4456	42991.9
278.554	0.283	0.792	77592.6
194.778	0.537	0.3895	37938.7
100.530	1.306	0.00946	10106.4
20.9439	1.847	0.00412	438.64

I^2	w*I	P^2	T (%)	• ANN Resl. T (%)
3.459	6692.3	0.00042	0.125	0.12442
0.690	11270.4	0.02722	0.3875	0.38885
0.173	7472.0	0.19855	0.625	0.62401
0.080	6213.6	0.62726	0.8375	0.83688
0.288	10937.7	0.15171	0.5875	0.58748
1.705	17237.5	0.00894	0.3	0.30165
3.411	1496.3	0.00001	0.0625	0.06311

5. CONCLUSIONS

It is considerably important success to access this performance increment as a basic study for further approximations of the subject, that is, under changing the value of the load, the load-torque rises to the same value performance in a very short time (mili seconds degree for CBA and nano seconds degree FBA). In the application of Standard Order-Classic Back-propagation Algorithm structure, as the input values to ANN, rotor speed (n) as angular speed (w), current (I) and power (P), as the output value to it torque (T) of the motor to increase the performance of the system are made and discriminated with very low error rate %0.0011 for 300000 iterations. These results are %0.0007 for 300000 iterations for Standard Order-Fast Back-propagation Algorithm structure, %0.0009 for 300000 iterations for High Order-Classic Back-propagation Algorithm structure and %0.0004 for 300000 iterations for High Order-Fast Back-propagation Algorithm structure. The results of the last one (High Order-Fast Back-propagation Algorithm Structure) are better than the others in case of the error ones.

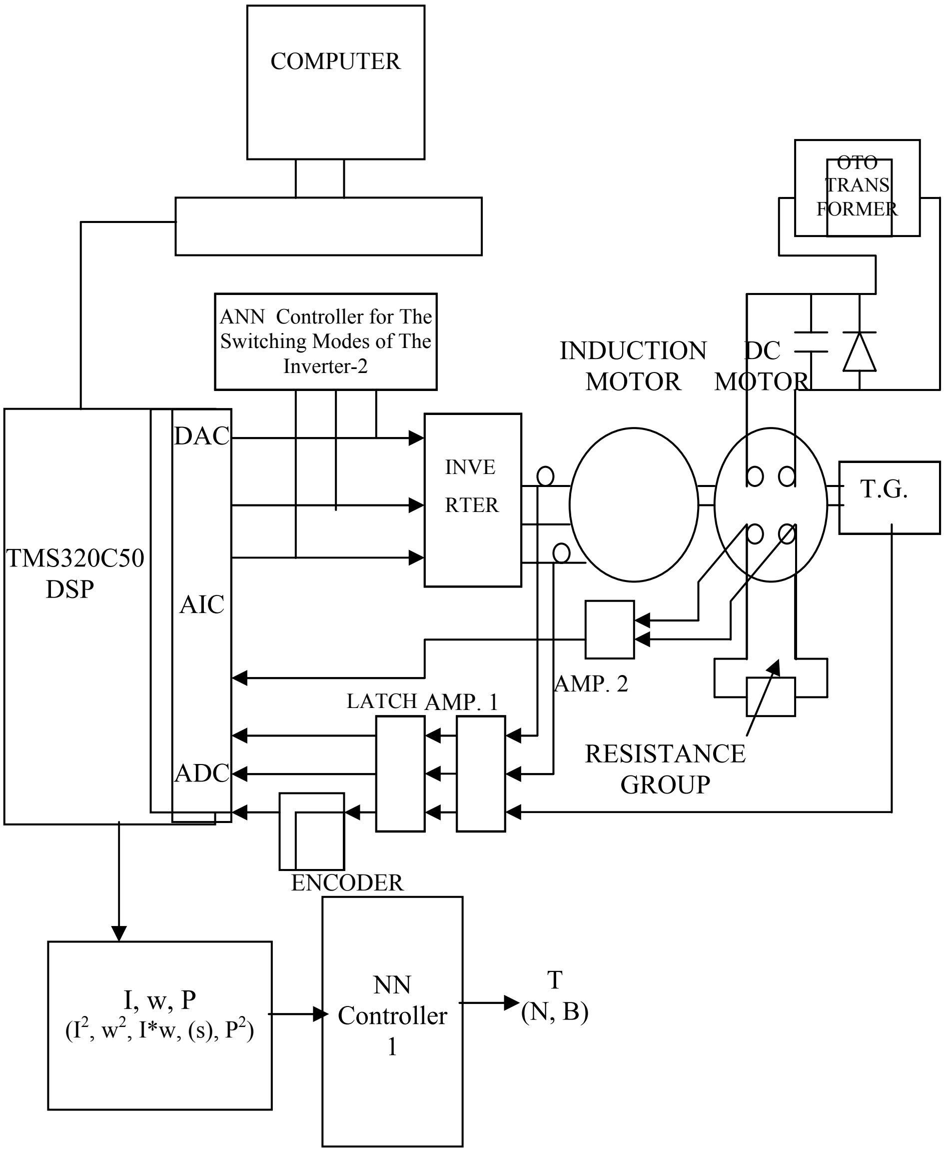


Figure 8. The block diagram of DSP- based control system including ANN controller

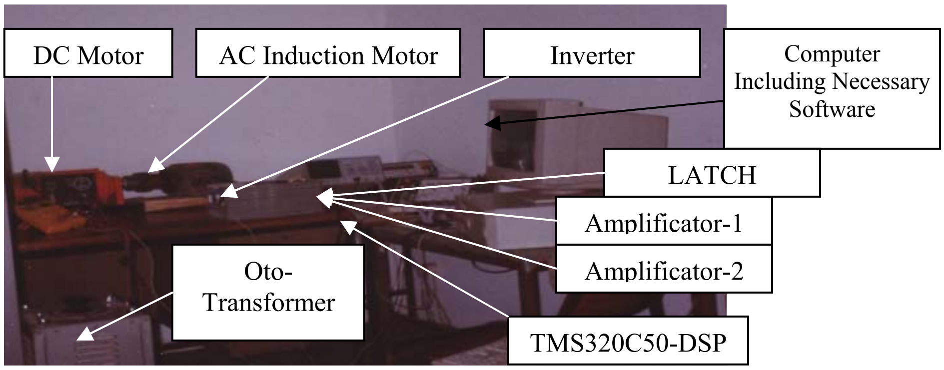


Figure 9. The photo of whole real-time induction motor control system for the standard 6-IGBT inverter

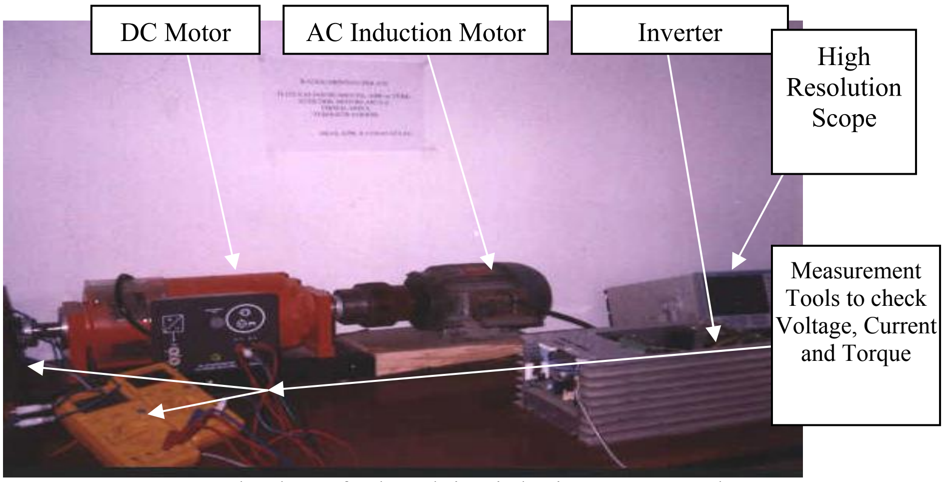


Figure 10. The photo of only real-time induction motor control system part

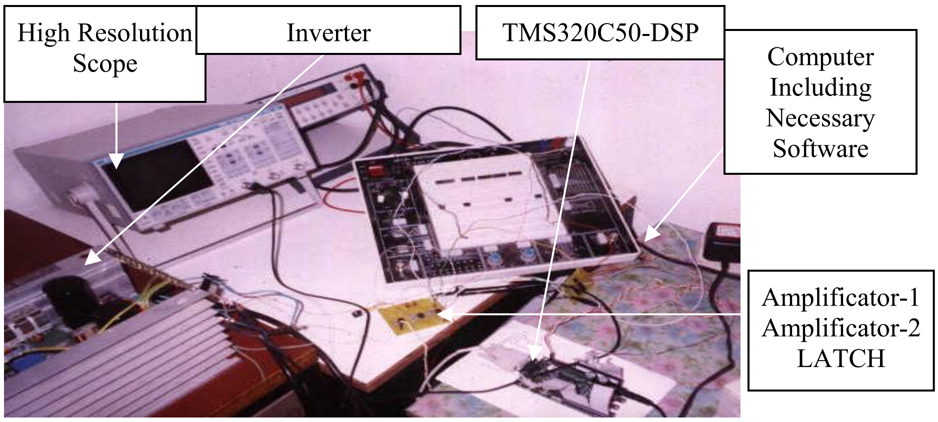


Figure 11. The photo of DSP (Digital Signal Processor) and amplificators part of the system

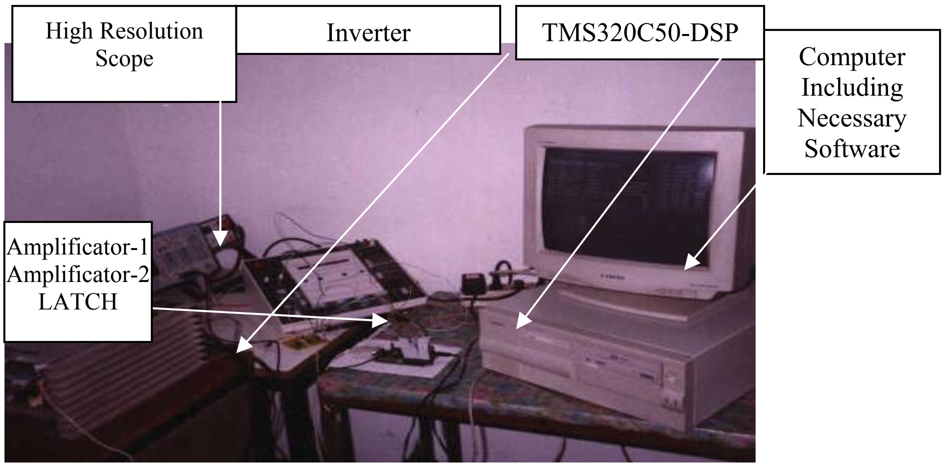


Figure 12. The photo of DSP and computational part of the system

ACKNOWLEDGMENTS

This study was supported TI Europe Branch, ABB Turkish Branch and Turkish Electrical Motors A.S. as equipment and technical document. Thus, special thank to TI Europe Branch, ABB Turkish Branch and Turkish Electrical Motors A.S.

REFERENCES

[1] Boldea, I. Ve Nasar, S.A., “Vector Control of AC Drives, CRC Press”, 1992.
[2] Bose, B.K., “Power Electronics and AC Drives”, Prentice-Hall, New Jersey, Charles, S., 1963, Electrical Control Systems in Industry, New York Mc-Grow-Hill, 1986.
[3] Dote, Y., “Servo Motor and Motion Control Using Digital Signal Processors”, Texas Instruments-Prentice Hall, Englewood Cliffs, New Jersey, 1990.
[4] Harashima, F., "Power Electronics and Motion Control A Future Perspective", Proc. of The IEEE, Vol.82, No.8, August, 1994.
[5] Leonhard, W., “Control of Electrical Drives”, Springer-Verlag.
[6] Levine, W.S., 1996, “The Control Handbook, CRC and IEEE Press”, 1991.
[7] Tzau, Y. "DSP-Based Robust Control of an AC Induction Servo Drive for Motion Control", IEEE Transactions on Control Systems Technology, Vol.4, No.6, 614-626, 1996.
[8] Nasar, S.A., “Handbook of Electric Machines”, Substitutes New York: McGraw Hill, 1987.
[9] Bose, B.K., Expert System, Fuzzy Logic and Neural Network Applications in Power Electronics and Motion Control, Proceedings of the IEEE, Vol.82, No.8, 1303-1323, 1994.
[10] Karlik, B., Gulez K., “The Performance Analysis of Induction Motors with Artificial Neural Networks (ANN)”, Proc. of IEEE 21st International Conference on Industrial Electronics, Control, and Instrumentation (IECON-95), vol. 2, pp. 1452-1455, 6-10 November, 1995, Orlando, USA.
[11] Gulez, K., "The Performance Increment of The Induction Motor by Using A DSP Based Control System Supported by ANN", Ph.D. Thesis, Y.T.U., Istanbul, 1999.

[12] Haykin, S., “Neural Networks”, Macmillan Publishing Company, New Jersey, 1994.
[13] Miller, W.T., Sutton, R.S., Werbos, P.J., “Neural Networks for Control”, Third Printing, 1992.
[14] Karayiannis, N.B. And Venetsanopoulas A.N., “Artificial Neural Networks- Learning Algorithms, Performance Evaluation and Applications”, Kluwer Academic Publishers, 161-195, 1994.
[15] Texas Instruments TMS320C50 User's Guide, 1997.