

An Indirect Adaptive Vector Control of the Induction Motor Velocity Using Neural Networks

*Ieroham Baruch**, *Irving Pavel de la Cruz A.**, *Ruben Garrido**,
*Boyka Nenkova***

**Department of Automatic Control, CINVESTAV-IPN, 07360 Mexico D. F., Mexico.*

*** IIT-BAS, 1113 Sofia, Bulgaria*

Abstract: *The paper proposes a neural network solution to the indirect vector control of three phase induction motor including a real-time trained neural controller for the IM angular velocity, which permitted the speed up reaction to the variable load. The basic equations and elements of the indirect field oriented control scheme are given. The control scheme is realized by one recurrent and two feedforward neural networks. The first one is learned in real-time by the dynamic BP method and the two FFNNs are learned off-line by the Levenberg-Marquardt algorithm with data taken by PI-control simulations. The final set up MSE of the LM algorithm is of 10^{-10} . The graphical results of modeling show a better performance of the adaptive NN control system with respect to the PI controlled system realizing the same computational control scheme with variable load.*

Keywords: *Modeling and simulation, induction motor, feedforward neural networks, Levenberg-Marquardt learning algorithm, field oriented control, indirect vector control.*

1. Introduction

The Neural Networks (NN) applications for identification and control of electrical drives became very popular in the last decade. In [1] an adaptive neuro-fuzzy system is applied for a stepping motor drive control. In [2] a multilayer perceptron-based-neural-control is applied for a DC motor drive. In [3] a recurrent neural network is applied for identification and adaptive control of a DC motor drive mechanical system. In the last decade a great boost is made in the area of induction motor drive control. The Induction Motor (IM), particularly the cage type, is most commonly used in adjustable speed AC drive systems [4]. The control of AC machines is considerably

more complex than that of DC machines. The complexity arises because of the variable-frequency power supply, AC signals processing, and complex dynamics of the AC machine [4, 5]. In the vector or Field-Oriented Control (FOC) methods, an AC machine is controlled as a separately excited DC machine, where the active (torque) and the reactive (field) current components are orthogonal and mutually decoupled so they could be controlled independently [4–7]. There exist two methods for PWM current controlled inverter – direct and indirect vector control, [4]. This paper will consider the indirect control method, where the slip angle, the direct and quadrature axes stator current set point components in stationary rotation frame are computed from the torque and rotor flux set points and used for vector control. There are several papers of NN application for AC motor drive indirect vector control. In [8] a Feedforward NN (FFNN) and Backpropagation (BP) learning are used for angular velocity estimation and control of an IM, using only the stator current measurements. The authors of [9] presented a method of NN velocity estimation and IM control based on the flux, voltage and currents models. In [10] a neural controller is implemented based on a TMS320C30 microprocessor in order to emulate an indirect Field Oriented Control (FOC) of an IM drive. In [11] an adaptive velocity controller is presented by a reference model, based on neural networks. In [12] a model referenced robust method of NN velocity control is proposed, that is based on neural identifier and a neural PI controller. In [13] a NN based adaptive control of an IM is proposed. The NN learning algorithm is derived using the Lyapunov theorem of stability. In [14] the authors proposed an IM velocity control scheme, containing a conventional PI controller, a dynamic compensator, a neural IM parameter identifier and a NN load torque estimator. The NN identifier is used to estimate the IM parameters and so to tune the dynamic compensator gains and the output signal of the NN load torque estimator is used for feedforward control. In [15] it is proposed to use a NN in order to design a self tuning PI velocity controller incorporated in an IM indirect vector control scheme. The paper [16] proposed to use a NN as an adaptive feedforward IM velocity controller. In [17] a FFNN-based estimator of the feedback signals is used for induction motor drive FOC system. The authors of [18] proposed two NN-based methods for FOC of induction motors. The first one used a NN flux observer in a direct FOC. The second one used a NN for flux and torque decoupling in an indirect FOC. The results and particular solutions obtained in the referenced papers show that the application of NN offers a fast and improved alternative of the classical FOC schemes. The present paper proposes a total neural solution of an indirect IM velocity vector control problem which assures fast response and adaptation to a variable load.

2. Models of the induction machine

2.1. A phase (a, b, c) model

The Induction Motor (IM) equations [6, 7], for stator and rotor voltages in vector-matrix form are given as

$$(1) \quad v_{abcs} = r_s i_{abcs} + p \lambda_{abcs} ,$$

$$(2) \quad v_{abcr} = r_r i_{abcr} + p \lambda_{abcr} ,$$

$$(3) \quad r_s = r_s I_3; r_r = r_r I_3,$$

where:

$$(4) \quad \begin{aligned} v_{abcs} &= (v_{as}, v_{bs}, v_{cs})^T; v_{abcr} = (v_{ar}, v_{br}, v_{cr})^T; \\ i_{abcs} &= (i_{as}, i_{bs}, i_{cs})^T; i_{abcr} = (i_{ar}, i_{br}, i_{cr})^T; \\ \lambda_{abcs} &= (\lambda_{as}, \lambda_{bs}, \lambda_{cs})^T; \lambda_{abcr} = (\lambda_{ar}, \lambda_{br}, \lambda_{cr})^T \end{aligned}$$

are voltage, current, and flux, stator and rotor, three dimensional (a, b, c) vectors, with given up phase components; r_s and r_r are stator and rotor winding resistance diagonal matrices, with given up equal elements r_s and r_r , respectively; I_3 is an identity matrix with dimension three, and p is a Laplacian differential operator. The vector-matrix block-form representation of the flux leakage is given by the equation

$$(5) \quad \begin{bmatrix} \lambda_{abcs} \\ \lambda_{abcr}^? \end{bmatrix} = \begin{bmatrix} L_{ss}^{abc} & L_{sr}^{2abc} \\ (L_{sr}^{2abc})^T & L_{rr}^{2abc} \end{bmatrix} \begin{bmatrix} i_{abcs} \\ i_{abcr}^? \end{bmatrix},$$

where the stator, rotor and mutual block-inductance 3×3 matrices are:

$$(6) \quad L_{ss}^{abc} = \begin{bmatrix} L_{ls} + L_{ss} & -L_{ss} / 2 & -L_{ss} / 2 \\ -L_{ss} / 2 & L_{ls} + L_{ss} & -L_{ss} / 2 \\ -L_{ss} / 2 & -L_{ss} / 2 & L_{ls} + L_{ss} \end{bmatrix},$$

$$(7) \quad L_{rr}^{2abc} = \begin{bmatrix} L_{lr}^? + L_{ss} & -L_{ss} / 2 & -L_{ss} / 2 \\ -L_{ss} / 2 & L_{lr}^? + L_{ss} & -L_{ss} / 2 \\ -L_{ss} / 2 & -L_{ss} / 2 & L_{lr}^? + L_{ss} \end{bmatrix},$$

$$(8) \quad L_{sr}^{2abc} = \left[L_{rs}^{2abc} \right]^T = L_{ss} \begin{bmatrix} \cos \theta_r & \cos[\theta_r + (2\pi / 3)] & \cos[\theta_r - (2\pi / 3)] \\ \cos[\theta_r - (2\pi / 3)] & \cos \theta_r & \cos[\theta_r + (2\pi / 3)] \\ \cos[\theta_r + (2\pi / 3)] & \cos[\theta_r - (2\pi / 3)] & \cos \theta_r \end{bmatrix}.$$

The matrix elements here are: L_{ls} , L_{lr} – stator and rotor leakage inductances; L_{ss} , L_{rr} – stator and rotor winding inductances. Using the winding turns stator/rotor ratio n , the relative leakage inductance L_{ls}' , could be written as

$$L_{lr}^? = n^2 L_{lr}; \theta_r = \int_0^t \omega_r(\xi) d\xi + \theta_r(0),$$

where θ_r and ω_r are the angular rotor position and the angular rotor velocity, respectively.

Now, the voltage equations (1) and (2) could be expressed with respect to the stator in the final (a, b, c) form

$$(9) \quad \begin{bmatrix} v_{abcs} \\ v_{abcr}^? \end{bmatrix} = \begin{bmatrix} r_s + pL_{ss}^{abc} & pL_{sr}^{abc} \\ (pL_{sr}^{abc})^T & r_r^? + pL_{rr}^{abc} \end{bmatrix} \begin{bmatrix} i_{abcs} \\ i_{abcr}^? \end{bmatrix},$$

where the relative rotor voltage, current, flux and resistance values are:

$$(10) \quad \begin{aligned} v_{abcr}^? &= nv_{abcr}; i_{abcr}^? = (1/n)i_{abcr}, \\ \lambda_{abcr}^? &= n\lambda_{abcr}; r_r^? = n^2 r_r. \end{aligned}$$

2.2. A ($q, d, 0$) model

The (a, b, c) model is very complicated for control, so it could be simplified using a transformation to the ($q, d, 0$) form. The AC motor equations for the stator and rotor voltages in vector-matrix form are given as follows:

$$(11) \quad v_{qd0s} = r_s i_{qd0s} + \Omega \lambda_{qd0s} + p \lambda_{qd0s},$$

$$(12) \quad v_{qd0r} = r_r i_{qd0r} + \Delta \Omega \lambda_{qd0r} + p \lambda_{qd0r},$$

where:

$$(13) \quad \begin{aligned} v_{qd0s} &= (v_{qs}, v_{ds}, v_{0s})^T; \quad v_{qd0r} = (v_{qr}, v_{dr}, v_{0r})^T, \\ i_{qd0s} &= (i_{qs}, i_{ds}, i_{0s})^T; \quad i_{qd0r} = (i_{qr}, i_{dr}, i_{0r})^T, \\ \lambda_{qd0s} &= (\lambda_{qs}, \lambda_{ds}, \lambda_{0s})^T; \quad \lambda_{qd0r} = (\lambda_{qr}, \lambda_{dr}, \lambda_{0r})^T \end{aligned}$$

are: voltage, current, and flux, stator and rotor, three-dimensional ($q, d, 0$) vectors, with given up components; r_s and r_r are stator and rotor resistance diagonal matrices, given by (3); Ω , and $\Delta \Omega$ are diagonal angular velocity matrices, given by

$$(14) \quad \Omega = \begin{bmatrix} \omega & 0 & 0 \\ 0 & -\omega & 0 \\ 0 & 0 & 0 \end{bmatrix}; \quad \Delta \Omega = \begin{bmatrix} \omega - \omega_r & 0 & 0 \\ 0 & -(\omega - \omega_r) & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

The vector-matrix block-form representation of the flux leakage is given by the equation

$$(15) \quad \begin{bmatrix} \lambda_{qd0s} \\ \lambda_{qd0r}^? \end{bmatrix} = \begin{bmatrix} L_{ss}^{qd0} & L_{sr}^{qd0} \\ (L_{sr}^{qd0})^T & L_{rr}^{qd0} \end{bmatrix} \begin{bmatrix} i_{qd0s} \\ i_{qd0r}^? \end{bmatrix},$$

where the stator, rotor and mutual block-inductance 3×3 matrices are:

$$(16) \quad L_{ss}^{qd0} = \begin{bmatrix} L_{ls} + L_{ss} & 0 & 0 \\ 0 & L_{ls} + L_{ss} & 0 \\ 0 & 0 & L_{ls} \end{bmatrix},$$

$$(17) \quad L_{rr}^{qd0} = \begin{bmatrix} L_{lr}' + L_m & 0 & 0 \\ 0 & L_{lr}' + L_m & 0 \\ 0 & 0 & L_{lr}' \end{bmatrix},$$

$$(18) \quad L_{sr}^{qd0} = \left[L_{rs}^{qd0} \right]^T = L_m I_3,$$

where L_m represents the mutual inductance.

The $(q, d, 0)$ model could be written in the stationary and synchronous frames taking the angular velocity equal to: $\omega = 0$ and $\omega = \omega_e$, where ω_e corresponds to the angular velocity of the stator field.

Now we could write the scalar electromagnetic torque equation which could be expressed in the following four forms:

$$(19) \quad T_{em} = \frac{3}{2} \frac{P}{2\omega_r} [\omega \lambda_{d-q,s}^T i_{q-d,s} + (\omega - \omega_r) \lambda_{d-q,r}^T i'_{q-d,r}],$$

$$(20) \quad T_{em} = \frac{3}{2} \frac{P}{2} \lambda_{d-q,s}^T i_{q-d,s},$$

$$(21) \quad T_{em} = \frac{3}{2} \frac{P}{2} \lambda_{q-d,r}^T i'_{d-q,r},$$

$$(22) \quad T_{em} = \frac{3}{2} \frac{P}{2} i_{d-q,r}^T i_{q-d,s},$$

where P is the number of poles and

$$\begin{aligned} \lambda_{d-q,s} &= (\lambda_{ds}, -\lambda_{qs})^T; & \lambda'_{d-q,r} &= (\lambda'_{dr}, -\lambda'_{qr})^T, \\ i_{q-d,s} &= (i_{qs}, i_{ds})^T; & i'_{q-d,r} &= (i'_{qr}, i'_{dr})^T, \\ i'_{d-q,r} &= (i'_{dr}, -i'_{qr})^T; & \lambda'_{q-d,r} &= (\lambda'_{qr}, \lambda'_{dr})^T. \end{aligned}$$

If we know the output power P_o of the IM, we could write the following relation for the torque with respect to the rotor angular velocity

$$(23) \quad T_{em} = \frac{P}{2}(P_o / \omega_r).$$

2.3. Field orientation conditions

The flux and torque equations decoupling must transform the stator flux, current and voltage vectors from (a, b, c) reference frame to $(q-d, s)$ reference frame and then to stationary and synchronous reference frame. Fig. 1 illustrates the current and voltage vector representations in stator and rotor synchronous frames. Fig. 1 illustrates also the magnetic field orientation, where the rotor flux vector is equal to the d -component of the flux vector, represented in a synchronous reference frame ($\lambda_{dr}^e = \lambda_r$), which is aligned with the d -component of the current in this frame. For more clarity, the current and flux orientation in the synchronous reference frame are shown on Fig. 2. So, the field orientation conditions are the following [7]:

$$(24) \quad \lambda_{qr}^e = 0; \quad p\lambda_{qr}^e = 0; \quad \lambda_r = \lambda_{dr}^e.$$

Taking into account that the rotor windings are shortcut (the rotor voltage is zero) and the field orientation conditions (24), the first two components of the equation (12), obtain the form

$$(25) \quad \begin{aligned} 0 &= r_r' i_{qr}^e + (\omega_e - \omega_r) \lambda_{dr}^e, \\ 0 &= r_r' i_{dr}^e + p \lambda_{dr}^e. \end{aligned}$$

From (15), for the q -component of the rotor flux, it is obtained:

$$(26) \quad \lambda_{qr}^e = L_m i_{qs}^e + L_r' i_{qr}^e = 0; \quad L_r' = L_{lr}' + L_m.$$

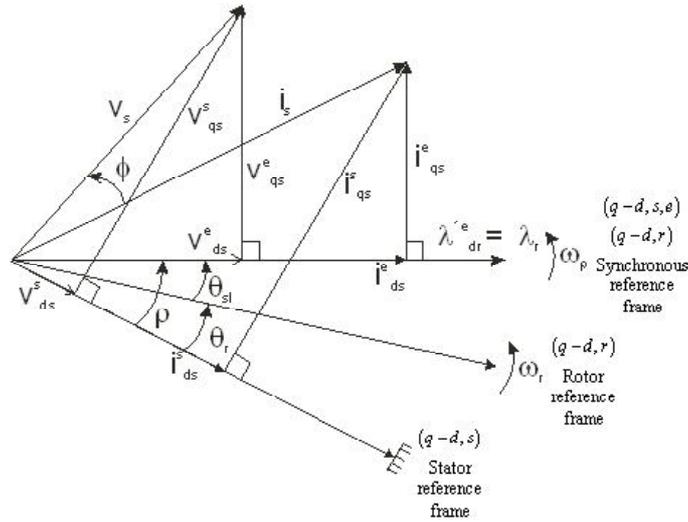


Fig. 1. The current and voltage vector representations in stator and in rotor synchronous reference frames

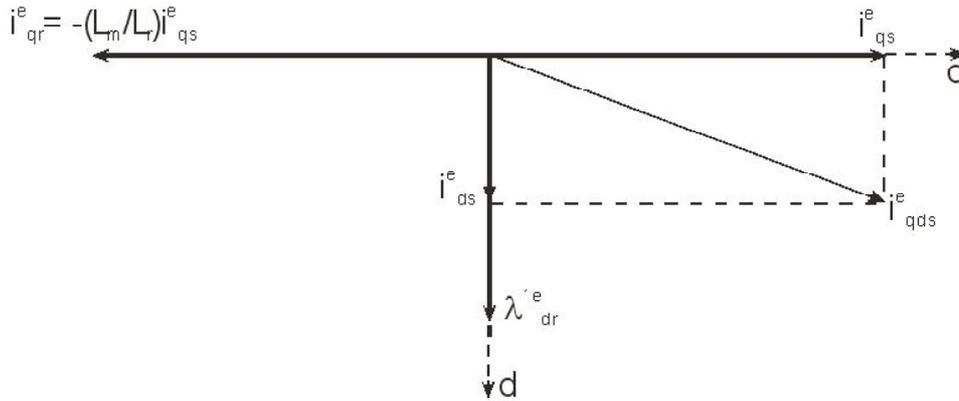


Fig. 2. The stator current and the rotor flux vector representations in synchronous reference frame

From (26) it is easy to obtain:

$$(27) \quad i_{qr}^e = -(L_m / L_r) i_{qs}^e .$$

Taking into account the condition (24), the torque equation (21) could be written in the form

$$(28) \quad T_{em} = -\frac{3}{2} \frac{P}{2} \lambda_{dr}^e i_{qr}^e .$$

The substitution of (27) in (28) finally gives

$$(29) \quad T_{em} = \frac{3}{2} \frac{P}{2} \frac{L_m}{L_r} \lambda_{dr}^e i_{qs}^e .$$

This equation shows that if the flux of the rotor is maintained constant, the torque could be controlled by the q -component of the stator current in synchronous reference frame. From the second equation of (25) it is easy to obtain the slipping angular velocity as:

$$(30) \quad \omega_e - \omega_r = -r_r' (i_{qr}^e / \lambda_{dr}^e) .$$

The substitution of (27) in (30) finally gives

$$(31) \quad \omega_e - \omega_r = (r_r' L_m / L_r) (i_{qs}^e / \lambda_{dr}^e) .$$

The final equations (29), (31) give the necessary basis for a direct decoupled field oriented (vector) control of the AC motor drive, where following Fig. 2, the q -component of the stator current produces torque and the d -component of the stator current produces flux.

2.4. Coordinate transformations

First of all we need to perform a coordinate transformation of stator variables from (a, b, c) to $(q-d, s)$ reference frames and its inverse. For sake of simplicity we shall show only the stator currents transformation – the other vectors transformations are similar to that, which is

$$(32) \quad \begin{bmatrix} i_{qs}^s \\ i_{ds}^s \end{bmatrix} = \begin{bmatrix} 1 & -1/2 & -1/2 \\ 0 & -\sqrt{3}/2 & -\sqrt{3}/2 \end{bmatrix} \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix}; \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_{qs}^s \\ i_{ds}^s \end{bmatrix}.$$

The $(q-d, s)$ to $(q-d, s, e)$ transformation of stator currents in synchronous reference frame and its inverse (see Fig. 1) are given by

$$(33) \quad \begin{bmatrix} i_{qs}^e \\ i_{ds}^e \end{bmatrix} = \begin{bmatrix} \cos \rho & -\sin \rho \\ \sin \rho & \cos \rho \end{bmatrix} \begin{bmatrix} i_{qs}^s \\ i_{ds}^s \end{bmatrix}; \begin{bmatrix} i_{qs}^s \\ i_{ds}^s \end{bmatrix} = \begin{bmatrix} \cos \rho & \sin \rho \\ -\sin \rho & \cos \rho \end{bmatrix} \begin{bmatrix} i_{qs}^e \\ i_{ds}^e \end{bmatrix}.$$

The combined stator current transformation from (a, b, c) to $(q-d, s, e)$ synchronous reference frame and its inverse are obtained combining equations (32) and (33), as

$$(34) \quad \begin{bmatrix} i_{qs}^e \\ i_{ds}^e \end{bmatrix} = \begin{bmatrix} \cos \rho & f_1 & f_2 \\ \sin \rho & f_3 & f_4 \end{bmatrix} \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix}; \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} = \begin{bmatrix} \cos \rho & \sin \rho \\ f_2 & f_4 \\ f_1 & f_3 \end{bmatrix} \begin{bmatrix} i_{qs}^e \\ i_{ds}^e \end{bmatrix},$$

$$(35) \quad \begin{aligned} f_1 &= [-(1/2)\cos \rho - (\sqrt{3}/2)\sin \rho], \\ f_2 &= [-(1/2)\cos \rho + (\sqrt{3}/2)\sin \rho], \\ f_3 &= [-(1/2)\sin \rho + (\sqrt{3}/2)\cos \rho], \\ f_4 &= [-(1/2)\sin \rho - (\sqrt{3}/2)\cos \rho]. \end{aligned}$$

2.5. Stator current set point estimation

The indirect control is based on equation (31). If this equation holds, this is a necessary and sufficient condition to produce an adequate field orientation. This assure that the d -flux rotor component in synchronous reference frame λ_{dr}^e will be aligned with the d -current stator component in synchronous reference frame i_{ds}^e (see Figs.1 and 2). Furthermore, this condition could be propagated to the set-point variables. The equation (31) could be expressed with respect to set point variables, so to obtain

$$(36) \quad \omega_e - \omega_r = (r_r' L_m / L_r')(i_{qs}^{e*} / \lambda_{dr}^{e*}).$$

In the same manner, from equation (29), written for the set-point variables, we could obtain a relationship for the q -current stator set-point component in synchronous reference frame i_{qs}^{e*} , expressed with respect to torque and flux set-points, as it follows:

$$(37) \quad i_{qs}^{e*} = (2/3)(2/P)(L_r' / L_m)(T_{em}^* / \lambda_{dr}^{e*}); \lambda_{dr}^{e*} = \lambda_r^*.$$

The computation of the d -current stator set-point component in synchronous reference frame i_{ds}^{e*} , required some more mathematical manipulations. From the rotor part of the equation (15) we could extract the equation for the d -flux rotor component which is

$$(38) \quad \lambda_{dr}^{'e} = L_r' i_{dr}^{'e} + L_m (i_{ds}^{'e} + i_{dr}^{'e}).$$

From (38) we could obtain:

$$(39) \quad i_{dr}^{'e} = (\lambda_{dr}^{'e} - L_m i_{ds}^{'e}) / L_r'.$$

From the second equation of (25) we could obtain the following equation:

$$(40) \quad i_{dr}^{'e} = -p \lambda_{dr}^{'e} / r_r'.$$

Equating the right parts of (40) and (39), and expressing the result with respect to the set-point variables, we could obtain the necessary d -current stator set-point component in synchronous reference frame i_{ds}^{e*} as follows:

$$(41) \quad i_{ds}^{e*} = (r_r' + L_r' p) (\lambda_{dr}^{'e*} / r_r' L_m).$$

If we accept that the rotor flux set-point (see (37)) is constant and its derivative is zero, the equation (41) is simplified as follows:

$$(42) \quad \lambda_{dr}^{'e*} = L_m i_{ds}^{e*}; \quad i_{ds}^{e*} = \lambda_{dr}^{'e*} / L_m.$$

The substitution of (42) in (36) gives

$$(43) \quad \omega_{sl} = \omega_e - \omega_r = (1 / \tau_r) (i_{qs}^{e*} / i_{ds}^{e*}); \quad \tau_r = (L_r' / r_r'); \quad \rho = \int_0^t (\omega_{sl} + \omega_r) ds.$$

So, the basic equations for an indirect FOC of IM are (37), (41) and (43).

3. Indirect vector control of the IM

3.1. A general control scheme

A general block diagram of the indirect vector control of the Induction Motor drive is given on Fig. 3. The indirect control scheme contains five principal blocks. They are: G_1 – block of angular velocity PI controller; block of the stator current set-point estimation. This computational block estimates the $(q-d, s, e)$ components i_{qs}^{e*} , i_{ds}^{e*} and the slip velocity ω_{sl} , performing field orientation (see equations (37), (41) and (43)); block C of ρ , $\sin \rho$, $\cos \rho$ computations (see equation (43)), where the slip velocity is added to the measured rotor velocity and the result is integrated on time (see (43)) to obtain the ρ -angle; block of coordinates $(q-d, s, e)$ to (a, b, c) current transformation (see equations (34), and (35)); block of the converter machine system, and induction motor. The block of the converter machine system contains a three phase bridge ASCII DC-AC current fed inverter, current hysteresis controllers, an induction motor model, and a model of the whole mechanical system driven by the IM, which is

$$(44) \quad (2/P)J(d\omega_r/dt) = T_{em} - T_L,$$

where J is the moment of inertia; T_L is the load torque.

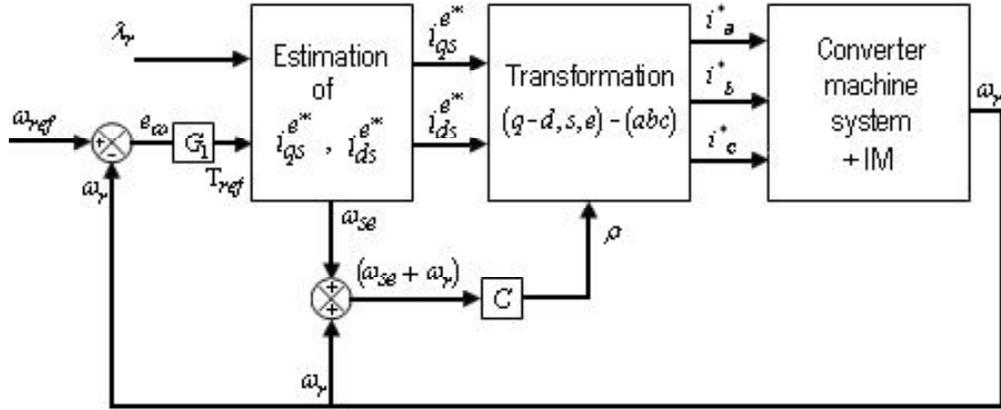


Fig. 3. General block-diagram of the indirect IM vector control

3.2. A Neural Network (NN) realization of the indirect vector control scheme

The simplified block-diagram of the direct neural vector control system, given on Fig. 3 is partly realized by NNs. It contains three NNs. We will describe in brief the function, the topology and the learning of each NN.

RNN1: The first Recurrent NN1 (RNN1) is an angular velocity recurrent neural controller with one input (the velocity error) and one output (the torque set point). The weights learning is done in real time using the Backpropagation (BP) algorithm. The topology (see Fig. 4), learning and stability proof of this RNN are fully described in [3].

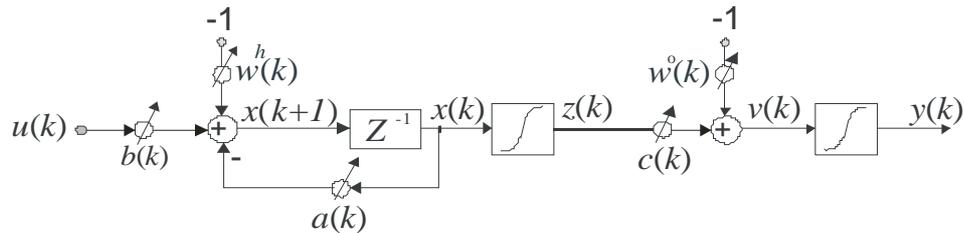


Fig. 4. Topology of the RNN1 velocity controller

The RNN1 function is given by

$$(45) \quad T^*(k+1) = \varphi\{c(k+1) \cdot \varphi[-a(k)x(k) + b(k)e_{vel}(k) - w^h(k)] - w^o(k+1)\},$$

where: $a(\cdot)$ and $b(\cdot)$ are hidden layer RNN1 weights; $c(\cdot)$ is an output layer RNN1 weight; $w^h(\cdot)$, $w^o(\cdot)$ are threshold weights of the hidden and output RNN1 layers, respectively; φ is a \tanh activation function; e_{vel} is a velocity error; T^* is the torque set point – output of the RNN1. The BP algorithm of learning for the output layer of the RNN1 [3] is given by

$$(46) \quad \begin{aligned} c(k+1) &= c(k) + \eta e_{\text{vel}}(k)[1 - (T^*(k))^2]z(k), \\ w^o(k+1) &= w^o(k) + \eta e_{\text{vel}}(k)[1 - (T^*(k))^2](-1). \end{aligned}$$

The BP algorithm of learning for the hidden layer of the RNN1 [3] is

$$(47) \quad \begin{aligned} a(k+1) &= a(k) + \eta R(k)x(k); b(k+1) = b(k) + \eta R(k)e_{\text{vel}}(k), \\ w^h(k+1) &= w^h(k) + \eta R(k)(-1); R(k) = c(k)e_{\text{vel}}(k)[1 - (z(k))^2], \end{aligned}$$

where the learning rate $\eta = 0.01$.

FFNN2: The second Feedforward NN2 (FFNN2) performs a current set points i_{qs}^* , i_{ds}^* and slip velocity ω_{sl} estimation, by means of a field orientation (see equations (37), (41) and (43)), given the torque and flux set points. If we admit that the flux set point is constant, we could use equation (42) instead of equation (41). The topology of this multilayer FFNN2 is of two inputs (flux and torque set points), three outputs (two current set points and slip velocity) and five and two neurons in the hidden layers (2-5-2-3). The off-line algorithm of its learning is the Levenberg-Marquardt (LM) one [19, 20]. The FFNN2 is learned by 2500 input-output patterns (half period) and generalized by another 2500 ones (the other half period) during 496 epochs. The final value of the MSE reached during the learning is of 10^{-10} .

FFNN3: The third NN3 performs a stator current set points ($q-d, s, e$) to (a, b, c) transformation (using equations (34), (35)). The FFNN3 topology has four inputs (two stator current set points $-i_{qs}^*, i_{ds}^*$; $\sin \rho, \cos \rho$), three outputs ($i_{as}^*, i_{bs}^*, i_{cs}^*$ – stator current set points) and two hidden layers of 30 and 10 neurons each (4-30-10-3). The FFNN3 learning is off-line, applying the Levenberg-Marquardt algorithm [19, 20]. The final value of the MSE reached during the learning is of 10^{-10} . The FFNN3 is learned by 2500 input-output patterns and generalized by 2500 ones during 30 epochs of learning.

4. Graphical results of control system modelling

The parameters of the IM used in the control system modeling are: power – 20 Hp; nominal velocity – $N = 1800$ Rev/m; pole number $P = 4$; voltage – 220 V; nominal current – 75 A; phase number 3; nominal frequency 60 Hz; stator resistance $r_s = 0.1062 \Omega$; rotor resistance referenced to stator $r_r' = 0.0764 \Omega$; stator inductance $L_s = 0.5689 \cdot 10^{-3}$ H; rotor inductance referenced to stator $L_r' = 0.5689 \cdot 10^{-3}$ H; magnetizing inductance $L_m = 15.4749 \cdot 10^{-3}$ H; moment of inertia $J = 2.8$ kg.m². The control system modelling is done changing the load torque in different moment of time. Figs. 5 and 6 show the angular velocity set point vs. the IM angular velocity without and with load torque changes. The results show that the angular velocity control system has a fast speed up response and satisfactory behaviour in case of a load change. Fig. 7 shows the flux graphics of the control system with PI control vs. neural control without and with load changes. The results show a faster and better response of the neural system. Figs. 8 and 9 show the graphics of the IM torque for a system with PI control vs neural control without and with load changes. The results show a faster and better response of the neural system in both of the cases.

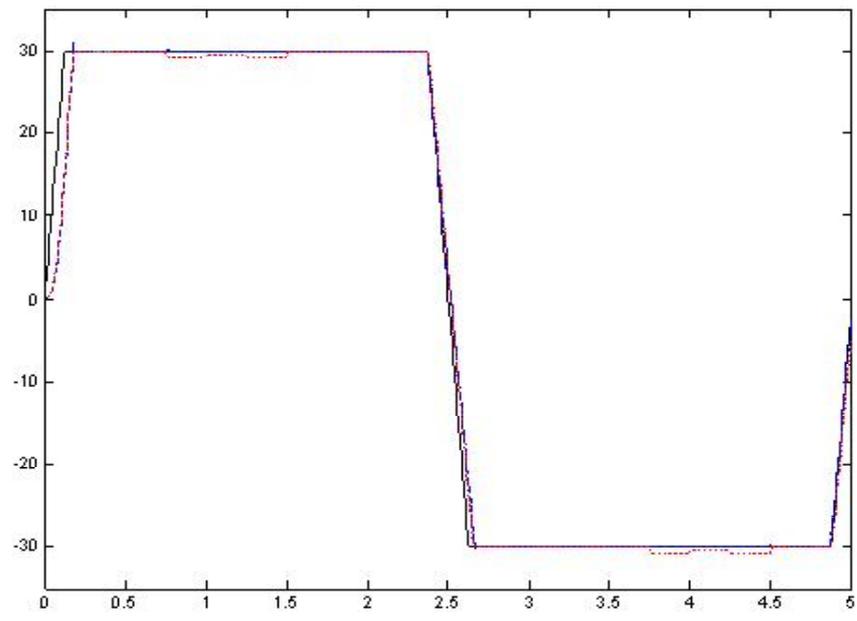


Fig. 5. General graphics of the angular velocity control with load variation in some regions

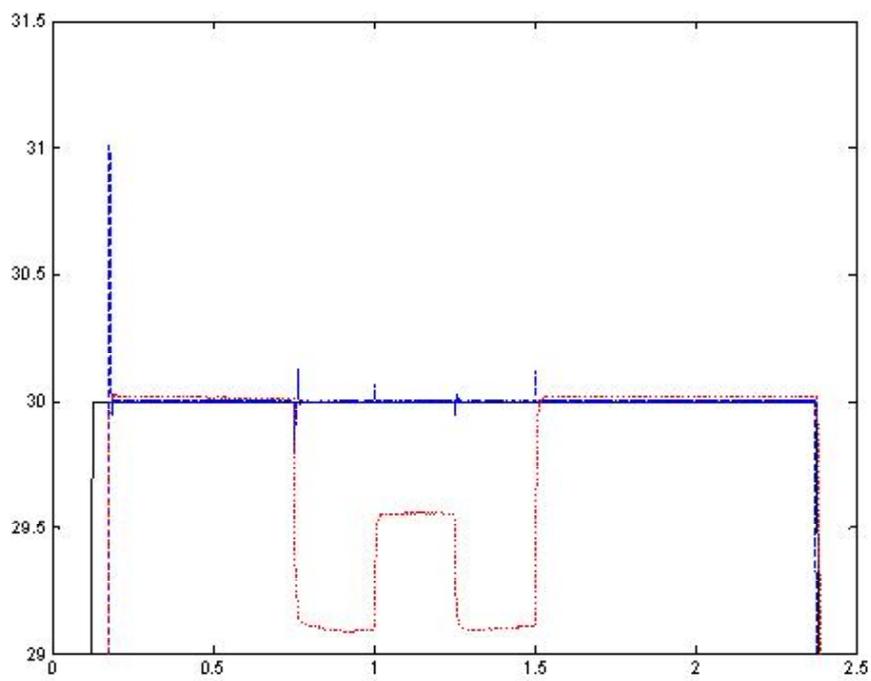


Fig. 6. Graphical results of angular velocity control with load changes (pointed line-graphics of the load torque variations)

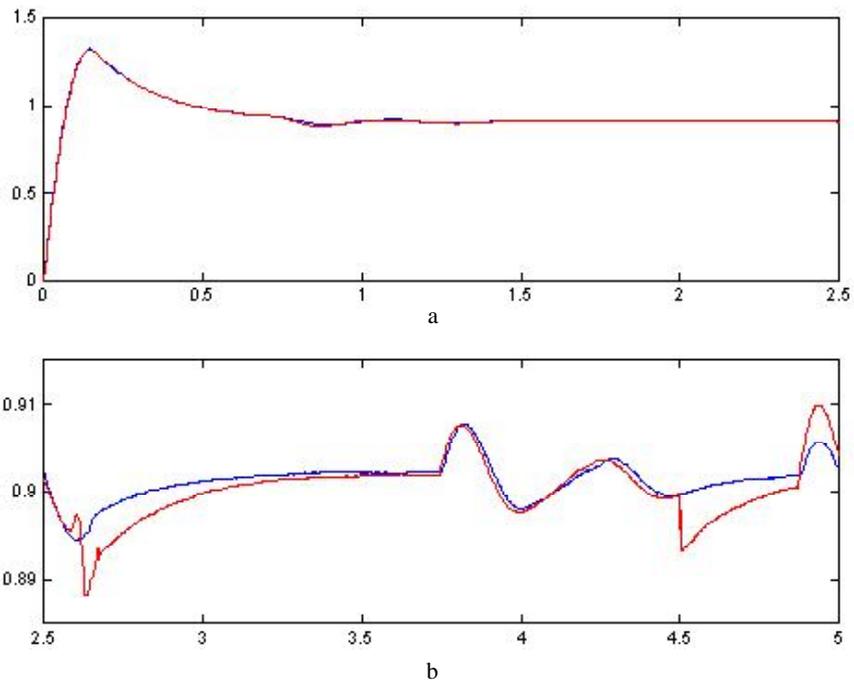


Fig. 7. Graphics of the flux control using both control schemes (classical control – continuous line; neural control – pointed line): a – systems start; b – load variations

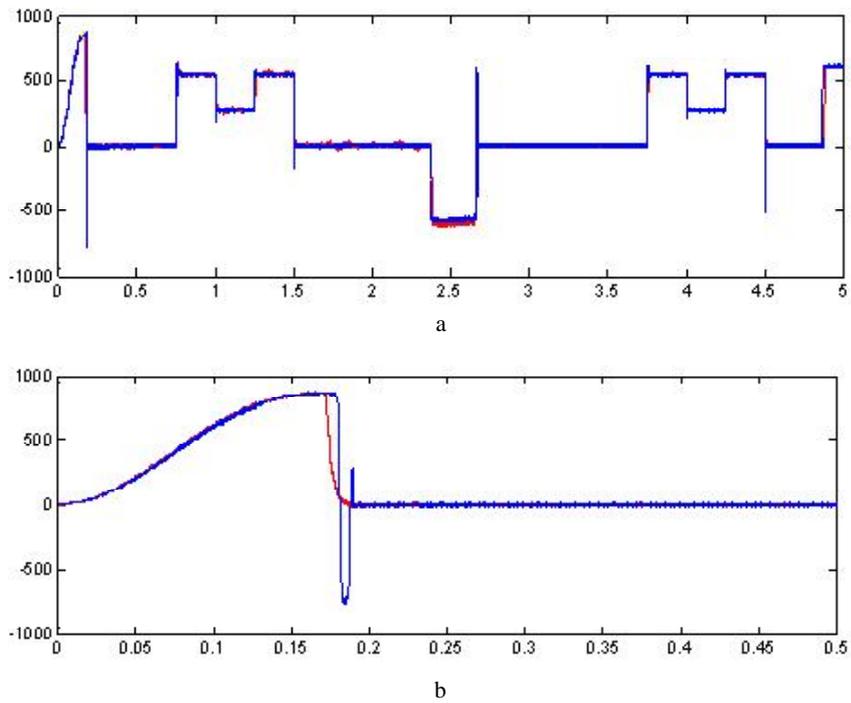


Fig. 8. Graphics of the torque control using both control schemes (classical control – continuous line; neural control – pointed line): a – process history; b – systems start

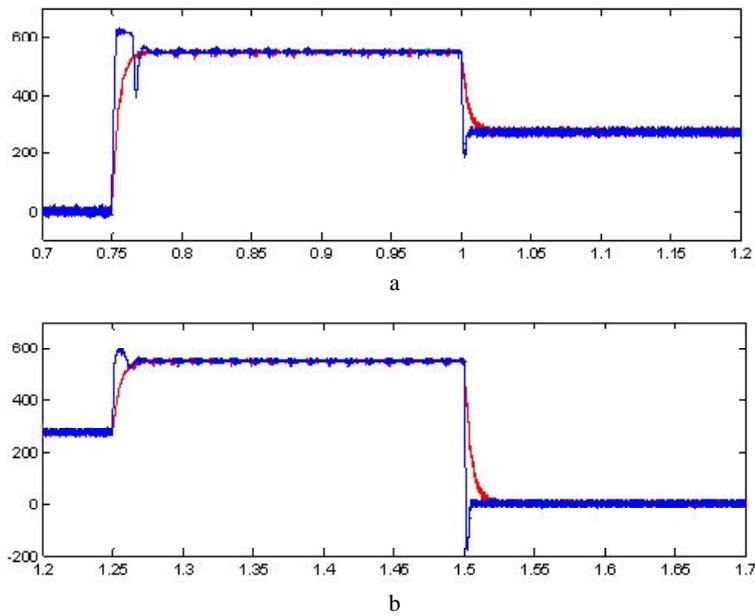


Fig. 9. Detailed graphical results of torque control in load variation conditions applying both control schemes (classical control-continuous line; neural control-pointed line): a – both torque graphics from 0.7 up to 1.2 s; b – both torque graphics from 1.2 up to 1.7 s

Fig. 10 shows the (a, b, c) stator current set-points and the (a, b, c) stator currents of current hysteresis controlled system start. Fig. 11 shows the same variables in the case of load changes. The results show a good performance of the neural control system at all.

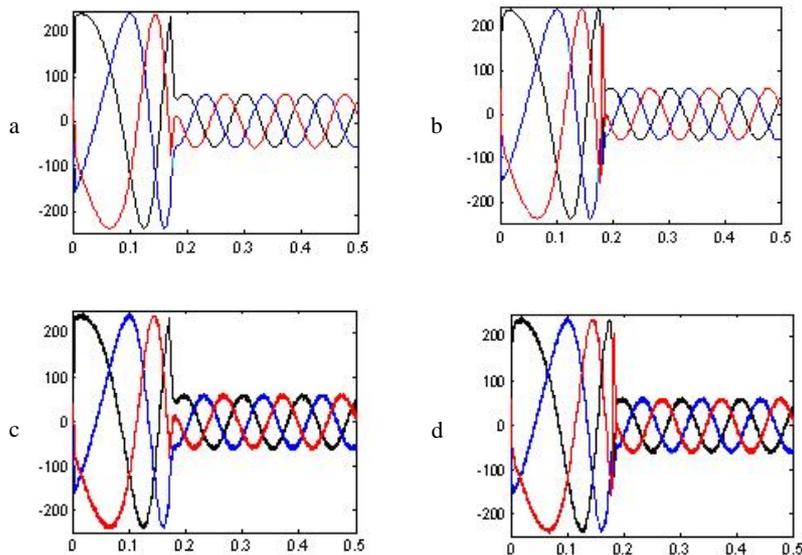


Fig. 10. Graphical results of (a, b, c) stator current set points (Figs. a and b) and (a, b, c) currents (Figs. c and d) during the start of the IM: a – current set points-PI control; b – current set points – neural control; c – currents-PI control; d – currents-neural control.

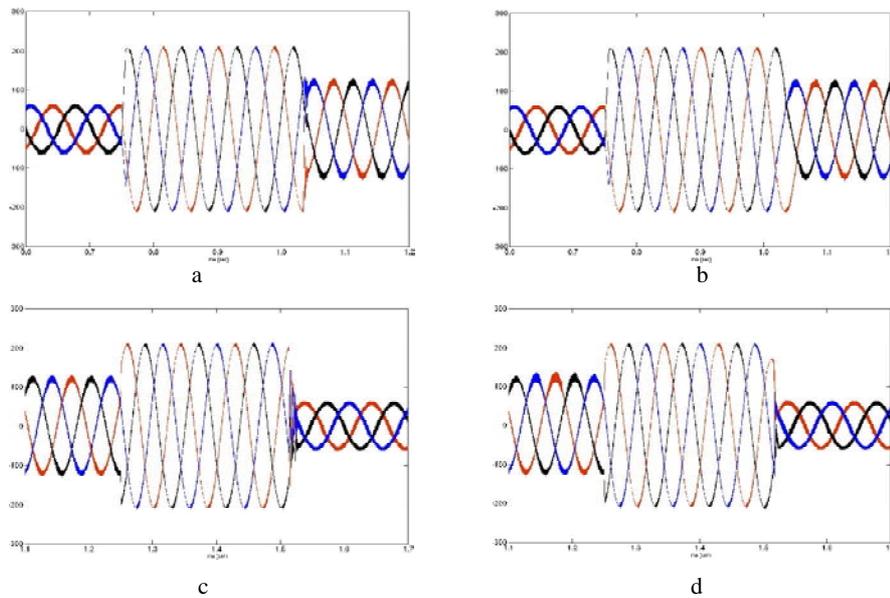


Fig. 11. Graphical results of (a, b, c) stator current set points (Figs. a and b and (a, b, c) currents (Figs. c and d during load variations: a – current set points-PI control; b – current set points-neural control; c – currents-PI control; d – currents-neural control.

5. Conclusions

The paper proposes a neural network solution to the indirect vector control of a three phase induction motor including a real-time trained neural controller for the IM angular velocity which permitted the speed up reaction to the variable load. The basic equations and elements of the indirect field oriented control scheme are given. The control scheme is realized by one recurrent and two feedforward neural networks. The first one is learned in real-time by the dynamic BP method and the two FFNNs are learned off-line by the Levenberg-Marquardt algorithm with data taken by PI-control simulations. The final set up MSE of the LM algorithm is of 10^{-10} . The graphical results of modelling shows a better performance of the adaptive NN control system with respect to the PI controlled system realizing the same computational control scheme with variable load.

References

1. Melin, P., O. Castillo. Intelligent Control of a Stepping Motor Drive Using an Adaptive Neuro-Fuzzy Inference System. – Information Science, Vol. **170**, December 2005, No 2-4, 133-151.
2. Weerasooriya, S., M. A. El-Sharkawi. Adaptive Tracking Control for High Performance DC Drives. – In: IEEE Trans. On Energy Conversion, Vol. **4**, Sept. 1991, No 3, 182-201.
3. Baruch, I. S., J. M. Flores, F. Nava R., I. R. Ramirez P., B. Nenkov a. An Advanced Neural Network Topology and Learning, Applied for Identification and Control of a D.C. Motor.– In: Proc. of the 1-st Int. IEEE Symposium on Intelligent Systems, September 2002, Varna, Bulgaria, ISBN 0-7803-7601-3, 289-295.

4. B o s e, B. K. Power Electronics and AC Drives. Prentice-Hall: Englewood Cliffs. New Jersey 07632, 1986, pp. 28-45, 264-291.
5. O r t e g a, R., A. L o r i a, P. N i c k l a s s o n, H. S i r a - R a m i r e z. Passivity-Based Control of Euler-Lagrange Systems. London, Berlin, Heidelberg, Springer Verlag, 1998.
6. C h e e - M u n, O n g. Dynamic Simulation of Electric Machinery. New York, Prentice Hall, 1998.
7. N o v o t n y, D. W., T. A. L i p o. Vector Control and Dynamics of AC Drives. New York, Oxford University Press, 1996.
8. B e n B r a h i m, L., S. T a d a k u m a, A. A k d a g. Speed Control of Induction Motor Without Rotational Transducer. – IEEE Transactions on Industry Applications, Vol. **35**, July/August 1999, No **4**, 844-850.
9. K i m, S. H., T. S. P a r k, J. Y. Y o o. Speed Sensorless Vector Control of an Induction Motor Using Neural Networks. – IEEE Transactions on Industrial Electronics, Vol. **48**, June 2001, No **3**, 609-614.
10. M o h a m a d i a n, M., E. N o w i c k i, F. A s h a f z a d e h, A. C h u, R. S a c h d e v a, E. E v a n i k. A Novel Neural Network Controller and its Efficient DSP Implementation for Vector-Controlled Induction Motor Drives. – IEEE Transactions on Industry Applications, Vol. **39**, November/December 2003, No **6**, 1622-1629.
11. S h y u, K u o K a i, H s i n g J a n g S h i e h, S h e n g S h a n g F u. Model Reference Adaptive Speed Control for Induction Motor Drive Using Neural Networks. – IEEE Transactions on Industrial Electronics, Vol. **45**, February 1998, No **1**, 180-182.
12. C h e n, T i e n C h i e, T s o n g T e r n g S h e u. Model Reference Neural Network Controller for Induction Motor Speed Control. – IEEE Transaction on Energy Conversion, Vol. **17**, June 2002, No **2**, 157-163.
13. C h i h M i n L i n, C h u F e i H s u. Neural-Network-Based Adaptive Control for Induction Motor Servomotor Drive System. – IEEE Transaction on Industrial Electronics, Vol. **49**, February 2002, No **1**, 115-123.
14. H u a n g, C h i c h Y i, T i e n C h i C h e n, C h i n g L e n g H u a n g. Robust Control of Induction Motor with Neural Network Load Torque Estimator and a Neural Network Identification. – IEEE Transactions on Industrial Electronics, Vol. **46**, October 1999, No **5**, 990-998.
15. S h e u, T s o n g T e r n g, T i e n C h i C h e n. Self-Tuning Control of Induction Motor Drive Using Neural Network Identifier. – IEEE Transactions on Energy Conversion, Vol. **14**, December 1999, No **4**, 881-886.
16. K u n g, Y. S., C. M. L i a w, M. S. O u y a n g. Adaptive Speed Control for Induction Motor Drives Using Neural Networks. – IEEE Transactions on Industrial Electronics, Vol. **42**, 1995, No **1**, 25-32.
17. S i m o e s, M. G., B. K. B o s e. Neural Network Based Estimation of Feedback Signals for a Vector Controlled Induction Motor Drive. – IEEE Transactions on Industry Applications, Vol. **31**, May-June 1995, No **3**, 620-629.
18. R a z z o u k, A. B a, A. C h e r i t i, G. O l i v i e r, P. S i c a r d. Field – Oriented Control of Induction Motors Using Neural-Network Decouplers. – IEEE Transactions on Power Electronics, Vol. **12**, July 1997, No **4**, 752-763.
19. H a g a n, M. T., M. B. M e n h a j. Training Feedforward Networks with the Marquardt Algorithm. – IEEE Trans. on Neural Networks, Vol. **5**, November 1994, No **6**, 989-993.
20. D e m u t h, H., M. B e a l e. Neural Network Toolbox User's Guide. Version 4. The Math Works, Inc. COPYRIGHT, 1992-2002.