A Neural Approach for Load Torque Identification in Electrical Machinery

ALESSANDRO GOEDTEL, IVAN NUNES DA SILVA, PAULO JOSE A. SERNI
Department of Electrical Engineering
State University of São Paulo – UNESP
CP 473, CEP 17033-360, Bauru, São Paulo
BRAZIL
agexp@uol.com.br, ivan@feb.unesp.br, paulojas@feb.unesp.br

Abstract: - The induction motors are largely used in several industry sectors. The dimensioning of an induction motor has still been inaccurate because in most of the cases the load behavior in its shaft is completely unknown. The proposal of this paper is to use artificial neural networks as tool for dimensioning of induction motors rather than conventional methods, which use classical identification techniques and mechanical load modeling. Simulation results are also presented to validate the proposed approach.

Key Words: - Induction motors, load modeling, parameter estimation, neural networks, system identification.

1 Introduction

The Three-phase Asynchronous Induction Motors (TAIM) are used in many industrial sectors as leading element to generate mechanic energy. Their main characteristics are based on robustness and low cost. However, when the load behavior is unknown, the choice procedure for a proper motor to a determined application becomes a difficult task, since the usual procedure is the experimental trial on the due application. If this motor presents current measures with a value over the nominal and speed under the admissible nominal value, it is clear then that the choice of this motor is improper. Next step, in this choice procedure, it is to substitute this motor by another with power addition in relation to the first one.

From the practical and mathematic analysis, it is demonstrated that the three-phase asynchronous induction motor, which are working in over dimensioned way, presents an increase of power factor and a decrease of efficiency. On the other hand, three-phase asynchronous induction motors working in an under dimensioned way present overheat and drastic reduction in its useful life.

A research made in CEMIG (Electric Energy Company of the Minas Gerais State - Brazil), with 3425 three-phase asynchronous induction motors, in several industrial sectors, has shown that 28.7% of them were over dimensioned and 5.9% of them worked under dimensioned.

Loads connected to the TAIMs shaft are pre-sorted according to the torque characteristic in four categories: constant, linear, quadratic and inverse [1]. This paper will consider only these four kinds of loading, which are mostly found in industrial applications.

The conventional methods used to determine torque are based on direct and indirect methods, as it was shown in [2]. According to the reference, the use of winding torquimeters brings the need of physical longitudinal displacing between the motor and the machine. The high start torque demanded by some loads requires over dimensioning of the sensor element, reducing thus its sensibility; the winding torquimeters must be carefully aligned to the motor shaft in order to avoid flexions that may reduce its useful life, making this system installation slow and expensive.

This paper proposes the use of ANN (Artificial Neural Networks) as alternative tool in the process of torque indirect measure in industrial loads with the objective of better dimensioning the induction motor, and also as optimization of control systems, whose variables of interest are the torque and the failure prediction in mechanic systems.

The organization of this paper goes on in the following order. In Section 2, it will be presented the mathematic modeling of the TAIM used in simulations. In Section 3, it will be described the basis of artificial neural networks. The results of simulation will be presented in Section 4 for the proposed model validation. In Section 5, it will be presented the conclusions of this work.

2 Mathematic Modeling of the Induction Motor

The mathematic model used in this work will simulate the behavior of the motor from its transient state until the steady state [3-6]. This model was developed by the use of the Matlab/Simulink software as computational tool. The machine parameters, such as: voltage, electric parameters of rotor and stator, moments of load and rotor inertia, and load torque are the input of the model. The electric current, the electromagnetic torque and the rotor speed are the outputs of the induction motor model. These variables will be used in the training process of the neural network.

2.1 Types and Characteristics of the TAIMs

The Three-phase Asynchronous Induction Motors are three-phase electrical machines widely used in industry, becoming them one of the main elements of mechanic traction.

Their main characteristics, such as robustness and low cost, make these motors preferred for most of the applications. To train an artificial neural network (ANN), whose objective is the torque estimation demanded by the load in the motor shaft, it is necessary the development of a mathematic model of this machine that considers the main characteristics of its non-linearities. However, due to its complexity, effects like temperature, copper and hysteresis losses will not be considered.

The three-phase asynchronous induction motors are classified in two basic categories, which as defined as follows: i) TAIM with rotor in squirrel-cage (TAIMSC), and ii) induction motors with coiled rotor (IMCR). All results obtained in this work are related to TAIMSC but may be expanded to the IMCR.

2.2 TAIM Mathematic Model

The objective of this section is the development of general mathematic model applicable to all three-phase asynchronous induction motors. The model must be capable of reproducing the electrical-mechanical behavior from the transient to the steady state since the results will be applied to the neural approach training. The detailed equation of the induction motor may be found in [3-6] and [14,15]. The simulation of the proposed model, using Matlab/Simulink, generated necessary data for the training process of the artificial neural network. Table 1 shows the induction motor parameters used in the simulation.

Table 1 - Induction motor specification and load parameters.

Standard Line – IV Poles – 60Hz – 220/380V	
Power (1 HP)	745.69 (W)
Stator Start Resistance	10.17 (Ω)
Stator Steady State Resistance	12.40 (Ω)
Rotor Start Resistance	5.80 (Ω)
Stator Start Inductance	1.77×10^{-2} (H)
Stator Steady State Inductance	2.05×10^{-2} (H)
Rotor Start Inductance	1.10×10^{-2} (H)
Rotor Steady State Inductance	4.84×10^{-2} (H)
Magnetizing Start Inductance	0.606 (H)
Magnetizing Steady State Inductance	0.546 (H)
Rotor Inertial Momentum	2.71×10^{-3} (Kg.m^2)
Load Inertial Momentum	8.13×10^{-3} (Kg.m^2)

2.3 Computational Model for Dynamic Simulation of Industrial Loads

The development of a computational mathematic model for load simulation has the goal of validating the proposed neural structure, which will be used in practical and real simulations. The proposed model is capable to provide to the TAIM shaft a resistant torque, which variation is related to the speed change. Figure 1 describes the proposed model.

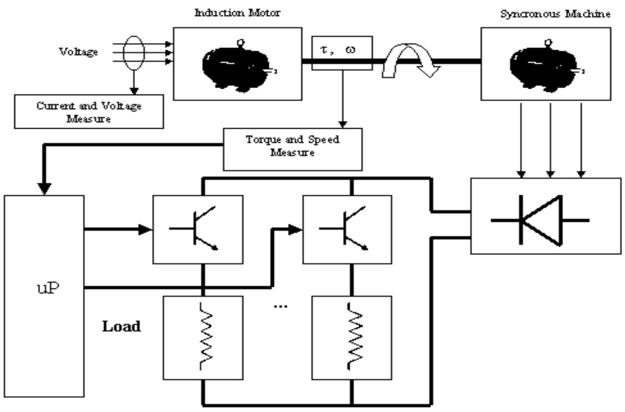

Fig. 1 - Computational model.

The induction motor is supplied through a three-phase network without the use of frequency inversors. The assembling synchronous machine-rectifier-loads will provide the resistant torque in its shaft. The micro controller that receives the signal from the optical encoder, which quantifies the speed of the motor shaft, determines setting the load in motion. The sequence of setting in motion differs for the four load types. For the constant load modeling all resistances must be linked together. In the other modeling the resistances will be set in action according to the desired load outline. In this work only the quadratic load will be simulated trough this mathematical model. The results obtained from this model simulation act as input to the neural network, and the torque measured in the shaft is compared to the output of the artificial neural network. This way, it is possible to validate the network computationally.

3 Using Artificial Neural Networks in Induction Motors

The use of artificial neural networks has shown efficient in solving a series of engineering and sciences problems. In this work, we have applied neural networks to estimate some parameters related to the induction machines.

Some approaches use neural networks for parameters estimation of electrical machines in feedback control of their speeds [7,8]. Another works make use of neural networks for fault prediction in induction motors in order to preventing maintenance, such as those described in [9,10]. It should also be noticed that the induction motor is the main energy consumption element in industries.

The appropriate specification of an induction motor requires the knowledge of that load to be coupled in its shaft. The lack of this information is compensated by the following procedure: current and speed are measured and if they have values out the range specified for that induction motor, then this motor will be substituted by another one, which is sometimes with 100% more powerful. It is also known that induction motors working over estimated increase the losses and they present a low power factor. As a consequence of all these facts, a big amount of electric energy will be lost.

Therefore, the main objective involved with this paper is in using artificial neural networks to estimate the load behavior on the motor shaft. In this work, it will be used a multilayer perceptron network, which has been trained by the backpropagation algorithm. This training algorithm has two basic steps: the first one, called propagation, applies values to the inputs of the ANN and it verifies the response signal in the output layer. This value is then compared with the desired signal in the output. The second step happens in the inverse way, that is, from the output to the input layer. The error produced by the network is used in the adjustment process of its internal parameters (weights and bias) [11].

The basic element of a neural network is the artificial neuron (Figure 2), which is also known either by node or processing element.

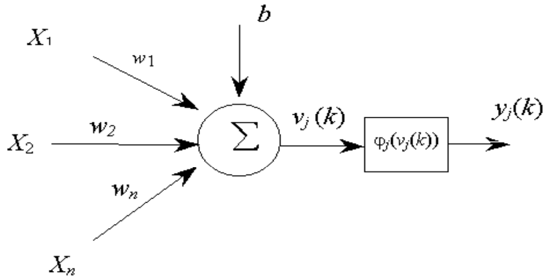

Fig. 2 - Representation of the artificial neuron.

The model of artificial neuron illustrated in Figure 2 can mathematically be modeled by the following equations:

$$v_j(k) = \sum_{i=1}^{n} X_i . w_i + b \tag{1}$$

$$y_j(k) = \varphi_j(v_j(k)) \tag{2}$$

where :

n is the number of input signals of the neuron.

X_i is the *i-th* input signal of the neuron.

w_i is the weight associated with the *i-th* input signal.

b is the threshold associated with the neuron.

$v_j(k)$ is weighted response (summing junction) of the *j-th* neuron with respect to the instant k.

$\varphi_j(.)$ is the activation function of the *j-th* neuron.

y_j(k) is the output signal of the *j-th* neuron with respect to the instant k.

Each artificial neuron is able to compute from input signals the respective output signal. The activation functions used to calculate the output signal are typically nonlinear. The neural networks that process analog data, which are also involved in this application, have often used activation functions of sigmoid type or hyperbolic tangent.

The adjustment process of the network weights (w_j) associated with the *j-th* output neuron is done from computation of the error signal with respect to the *k-th* iteration or *k-th* input vector (training example). This error signal is provided by the following equation:

$$e_j(k) = d_j(k) - y_j(k) \tag{3}$$

where $d_j(k)$ is the desired response to the *j-th* output neuron.

Adding all squared errors produced by the output neurons of the network with respect to *k-th* iteration, we have:

$$E(k) = \frac{1}{2} \sum_{j=1}^{p} e_j^2(k) \tag{4}$$

where p is the number of output neurons.

For an optimum weight configuration, $E(k)$ is minimized with respect to the synaptic weight w_{ji} . Therefore, the weights associated with the output layer of the network are updated using the following relationship:

$$w_{ji}(k) \leftarrow w_{ji}(k) - \eta \frac{\partial E(k)}{\partial w_{ji}(k)} \tag{5}$$

where w_{ji} is the weight connecting the *j-th* neuron of the output layer to *i-th* neuron of the previous layer, and η is a constant that determines the learning rate of the backpropagation algorithm.

The adjustment of weights belonging to the hidden layers of the network is also done in analogous way. The necessary steps to adjust these weights associated with the hidden neurons can be found in [11].

3.1 The LVQ-1 Network

In this paper the identification of the several load behavior was made using a network of the LVQ-1 type. A supervised algorithm, which from informations about the several classes that define the analyzed process, moves the quantizer vectors of the system with the objective of improving the decision regions of the system.

An input vector $\boldsymbol{x}_1$, which components have random values obtained from the input space, is compared with vectors $\boldsymbol{w}_j$ that represent the *j-th* classes (neurons) associated to the analyzed process. If the input vector $\boldsymbol{x}_i$ has higher proximity level in relation to the $\boldsymbol{w}_j$ vector, then the $\boldsymbol{w}_j$ vector is attracted towards the $\boldsymbol{x}_i$ vector; otherwise, vector $\boldsymbol{w}_j$ is repelled in relation to the direction of the vector $\boldsymbol{x}_i$. Since $C_{\boldsymbol{w}j}$ represents the associated class to vector $\boldsymbol{w}_j$ and $C_{\boldsymbol{x}i}$ demonstrates the associated class to the input vector $\boldsymbol{x}_i$, then the learning algorithm of the LVQ-1 network (applied only to the winner neuron) can be synthesized in the following form:

$$\text{if } C_{\boldsymbol{w}_j} = C_{\boldsymbol{x}_i}$$

$$\text{then } \boldsymbol{w}_j(\text{t}+1) = \boldsymbol{w}_j(t) + \eta[\boldsymbol{x}_i - \boldsymbol{w}_j(t)]$$

$$\text{else } \boldsymbol{w}_j(\text{t}+1) = \boldsymbol{w}_j(t) - \eta[\boldsymbol{x}_i - \boldsymbol{w}_j(t)]$$

where η is the learning rate. The convergence of vectors $\boldsymbol{w}_j$ occurs after the successive application of all input vectors $\boldsymbol{x}_i$ belonging to the respective input set. A detailed study of the LVQ-1 algorithm may be found in [11, 12].

The LVQ-1 network was trained for three situations: i) to identify type of torque and classify it between higher than zero or equal to zero; ii) to identify type of torque and classify it between constant and quadratic; and iii) to identify type of torque and classify it between linear and quadratic.

For all situations, 10 neurons (quantizer vectors) are used to represent each one of the classes, which represent the system tendencies. For the training, it was used 5 complete curves that describe phases of different behaviors of the current in the induction motor (from the start to the permanent state) in relation to the four types of load. After convergence of the quantizer vectors, the network is capable to identify the possible tendency of load when a new set of vectors, which describe the present behavior of the current, is presented. On next section will be presented simulation examples to validate the proposed approach.

3.1.1 Training of the Patterns Classifying Network

The patterns classifying network has the objective of receiving the current data vector and to classify it in one of the four types of load: linear, quadratic, constant or inverse.

The input data is the current that supplies the motor. Network 1 classifies the current data vector in initial torque higher than zero or equal to zero. In a second stage the same vector is presented to the second network, which depends on the initial classification, putting in action network 2 or network 3. Network 2 classifies the data vector in constant or inverse torque, while network 3 classifies it in quadratic or linear torque. Figure 3 illustrates the architecture described for patterns classification.

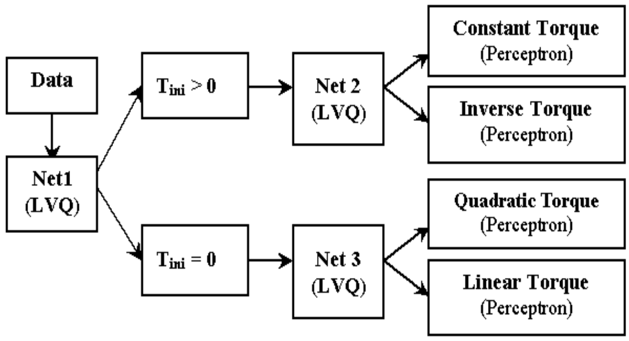

Fig. 3 - Patterns classifying network.

3.2 Interaction between LVQ Network and Multilayer Perceptron

The multilayer perceptron network was trained with inputs of voltage, current, speed and output of electromagnetic torque demanded by the load in the motor shaft. The mathematic simulations of the induction motor provided the data for the net training.

Each type of load had a trained perceptron network, that is, one network generalizes the loads of quadratic type, other for linear type, other for the constant, and finally, other for the inverse type.

To determine which perceptron network type must be put in action for generalization, we use a system of patterns recognition based on LVQ-1 networks.

4 Results of Simulation

The results of simulation obtained in this work were for a TAIM of 1 HP and 0.16 HP, modeled and implemented using Matlab/Simulink. The model and the procedure hereby applied may be used for other powers. In all trainings it was used the method proposed by Levenberg-Marquardt, as presented in [13]. The input signals of the network were the following ones: voltage (V), current (A) and rotor speed (rad/s). Each motor simulation has its own trained neural structure (LVQ + Perceptron).

The output signal calculated by the network is the torque. On the figures are presented the desired torque values and so the results obtained in the network output. This way it is possible to visualize the desired value and the value obtained in the network output in the same chart. In Figure 4, it is presented the results estimated by ANN for a load with constant load torque on 0.16 HP motors, which are largely used in conveyor belts.

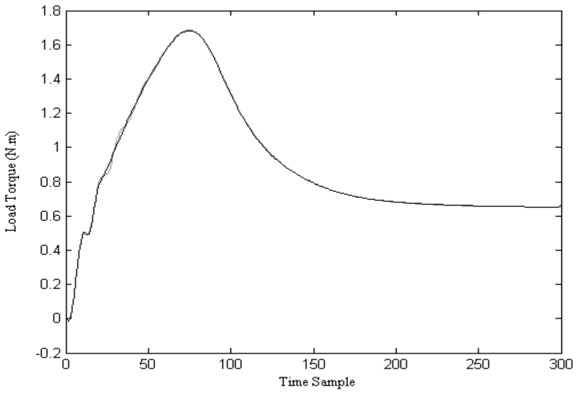

Fig. 4 - Estimation of torque to constant load.

In Figure 5, it is presented the result obtained from the estimation of the inverse torque in milling cuter and mandrellers for the 0.16 HP induction motor.

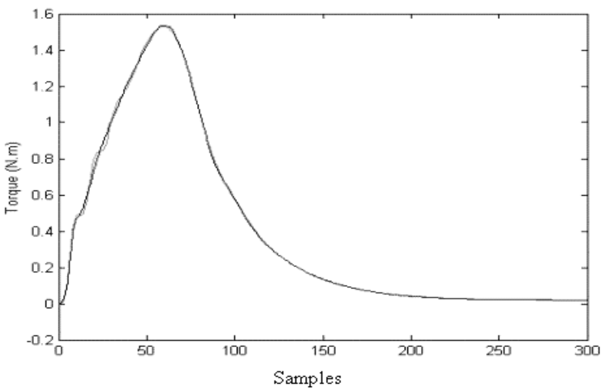

Fig. 5 - Estimation of torque to inverse load.

In Figure 6, it is presented the result of simulation for the load with quadratic behavior in the 1 HP TAIM, characteristic of pumps and ventilators.

In Figure 7, it is presented the result of simulation for the load with linear behavior applied in the 0.16 HP Motor.

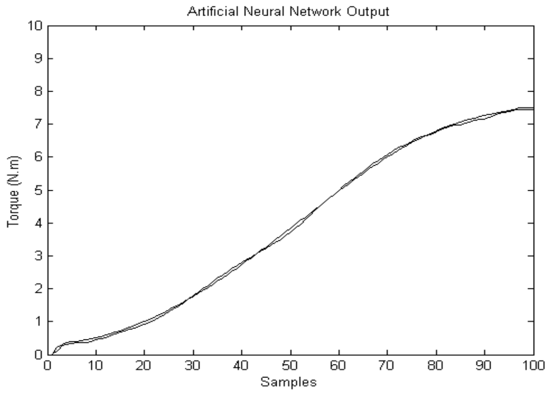

Fig. 6 - Estimation of torque to constant load.

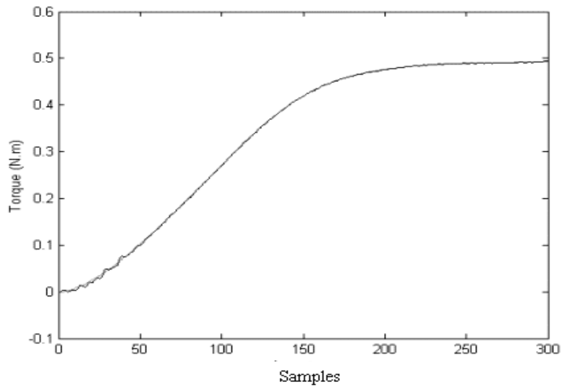

Fig. 7 - Estimation of torque to linear load.

In Figure 8, it is presented the results of simulation of the computational model for dynamic simulation of industrial loads, as presented in Section 2.3. The load behavior was imposed to the TAIM shaft trough the resistance switching sequence. It must be observed that the load was divided in twelve discrete steps. This switching procedure introduces distortion on current signal, resulting in error among the desired result and the ANN output only during the transient state.

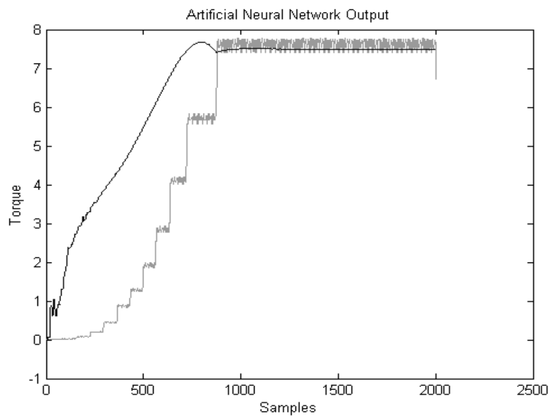

Fig. 8 - Estimation of torque to dynamic load.

The results of simulation confirm that it is possible to estimate torque in the TAIM shaft using artificial neural networks, which inputs are measures of voltage, current and speed. The perceptron network used in this simulation has a hidden layer with 25 artificial neurons.

In performed simulations, the errors found between desirable values and those presented in the network output do not surpass 5%. The classification adroitness of the LVQ-1 network was of 75% for loads with inverse and constant torque characteristics and agreement of 85% in the classification of linear and quadratic loads.

When the load behavior is known it is not necessary the use of pattern classifying architecture. In this case are used only perceptron networks to determine the torque behavior demanded by the axle from the start to the steady state. To improve the generalization results on dynamic load simulation it is strongly recommended the use of harmonic distortion during the ANN training.

5 Conclusion

This paper describes the application of artificial neural nets in torque estimation of loads mainly used in industrial premises, enabling to supply control systems and to help better dimensioning of a TAIM, as to put it in action for determined application. The results of simulation are considered satisfactory. The methodology used in this work may be applied to other loads and other types of motors.

The main justifications for the application are the following: i) the torque estimation is directly made by an artificial neural network, ii) the method may be used as an auxiliary tool in the control and dimensioning of a TAIM, iii) the method contributes to verify if a motor is overestimated or under dimensioned. The method here developed contributes in a significant way to the reduction of electric energy losses and the increase of the power factor, which are derived from bad dimensioning of electrical motors.

6 Acknowledges

To FAPESP (process 01/06266-5) by the financial support and to WEG Motors S.A. for providing parameters related to the Induction Motor.

References:

[1] L. P. C. Dias and O. S. Lobosco, *Electrical Motors: Selection and Application*, Vol. I, McGraw-Hill, 1988.

[2] R. M. Brito, *Electro-Electronic Torque Measuring System Applied to Rotating Devices Using Electrical Resistant Extensometers*, Doctorate Thesis (in Portuguese), Federal University of Rio Grande do Sul State (UFRGS/PPGEMM), 1994.

[3] E. Avolio, *A Contribution to Study of Thermal and Electric Behavior in Induction Motors*, Doctorate Thesis (in Portuguese), State University of Campinas (UNICAMP/FEEC), Brazil, 1992.

[4] C. M. Ong, *Dynamic Simulation of Electric Machinery Using Matlab/Simulink*, Prentice Hall, 1997.

[5] R. Reginatto and J. E. N. Rico, *Induction Motors Modeling*, UFSC, 1993.

[6] A. E. Fitzgerald, C. Kingsley and A. Kusko, *Electric Machiner*y, McGraw-Hill, 1975.

[7] F. -J. Lin, J. -C. Yu and M. -S. Tzeng, Sensorless Induction Spindle Motor Drive Using Fuzzy Neural Network Speed Controller, *Electric Power Systems Research*, Vol. 58, 2001, pp. 187-196.

[8] D. Kukolj, F. Kulic and E. Levi, Design of Speed Controller for Sensorless Electric Drives Based on AI Techniques: a Comparative Study, *Artificial Intelligence in Engineering*, Vol. 14, 2000, pp. 165-174.

[9] S. Kolla and L. Varatharasa, Identifying three-phase induction motor faults using artificial neural networks, *ISA Transactions*, Vol. 39, No. 4, 2000, pp. 433-439.

[10] D. K. Chaturvedi, P. S. Satsangi and P. K. Kalra, New Neuron Models for Simulating Rotating Electrical Machines and Load Forecasting Problems, *Electric Power Systems Research*, Vol. 52, 1999, pp. 123-131.

[11] S. Haykin, *Neural Networks - A Comprehensive Foundation*, Prentice Hall, 1999.

[12] A. A. C. de Freitas and I. N. da Silva, Technical Analyze of Stock Market Titles Using Artificial Neural Networks, *Brazilian Congress on Neural Networks*, 1999.

[13] M. T. Hagan and M. B. Menhaj, Training Feedforward Networks with the Marquardt Algorithm, *IEEE Trans. on Neural Networks*, Vol. 5, No. 6, 1994, pp. 989-993.

[14] A. Goedtel, I. N. da Silva, P. Serni and E. Avolio, Load Torque Estimation in Induction Motors Using Artificial Neural Networks, *IEEE International Joint Conference on Neural Networks*, 2002, pp. 1379-1384.

[15] A. Goedtel, I. N. da Silva, P. Serni and E. Avolio, A Neural Approach to Identification of Torque in Induction Motor Shaft, *V IEEE Industry Conference Applications (IEEE Induscon)*, 2002.