



# Speed Estimation of an Induction Motor with Sensorless Vector Control at Low Speed Range Using Neural Networks

Martin Gallo Nieves, Jorge Luis Diaz Rodriguez<sup>(✉)</sup>,  
and Jaime Antonio Gonzalez Castellanos

Universidad de Pamplona, Pamplona, Colombia  
martingallo6@gmail.com, jdiazcu@gmail.com,  
gcjaime@gmail.com

**Abstract.** This paper deals with the study, design and implementation of a speed estimation technique, using an Artificial Neural Network (ANN), for a three-phase induction motor type Squirrel Cage. The main objective of this technique is to estimate the speed of the engine from its own conditions at low speed, with the purpose of applying vector control without the use of a traditional speed sensor that supplies the control loop with the speed of rotation the motor. When replacing the traditional sensor with RNA, a sensorless control loop is obtained. The training of the RNA was done collecting data in different ranges of engine speed, both empty and loaded. The simulation of the system is carried out in the Simulink platform, from the Matlab software package and the implementation of the control strategy is done using the Raspberry Pi 3B card.

**Keywords:** Induction motor · Speed estimation · Vector control · Sensorless control · Artificial neural networks

## 1 Introduction

Squirrel-cage three-phase electric motors are the most commonly used in industrial processes for mechanical power supply. This is due to its characteristics compared to other types of electric motors, such as: lower weight, low cost, low physical robustness (weight/power ratio) and fewer parts [1].

However, a large part of the equipment and applications used in the industry must operate at variable speeds and controlled in an agile and precise way for its good performance.

Within the control strategies of the induction motors and especially the indirect vector control it is necessary to measure the rotor speed. This measurement was made from analog speed meters that present noise problems. Then appeared digital speed meters such as encoders, resolvers and photoelectric meters that allowed to solve most of the disadvantages of analog sensors but with a higher cost.

In the present work, a speed estimation strategy based on artificial intelligence is used, using neural networks that achieve excellent results by applying a novel strategy based on the use of delays in the data, applying it to a Sensorless vector control algorithm.

## 2 Induction Motor

If a three-phase voltage is applied to an induction motor, three three-phase currents will circulate; these currents produce a magnetic field  $B_s$  that rotates counterclockwise [1].

The rotation speed of the magnetic field is given by the following equation:

$$n_s = 120f_e/p \quad (1)$$

Where:

$n_s$ : speed of the magnetic field [revolutions per minute or simply *rpm*].

$f_e$ : frequency of the input voltage (power source) [Hz].

$p$ : number of motor poles.

### 2.1 Induction Motor Analysis

Figure 1 shows a diagram of a three-phase induction machine. This type of machines has cylindrical magnetic structures in both the rotor and the stator. These structures are the same in each of the parts of the electric motor, with a phase difference between them [2]. Additionally, the cage rotor induction machine is also symmetrical, for the same reasons. However, the number of rotor phases is greater than three. In fact, each bar of the cage constitutes a phase.

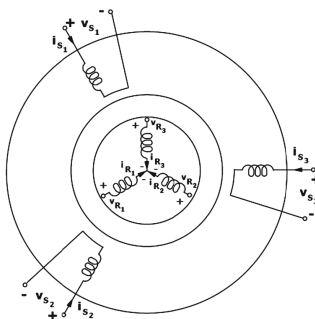


Fig. 1. Schematic representation of a three-phase induction motor.

### 2.2 Mathematical Model of the Three-Phase Induction Motor

The voltage model of a induction motor is defined by the sum of the voltage drop plus the magnetic flux equations. It is represented through the following equations [2]:

$$V_S = R_S I_S + L_{SS} \frac{dI_S}{dt} + L_{SR}(\theta) \frac{dI_R}{dt} + \frac{\partial L_{SR}(\theta)}{\partial \theta} I_R \frac{d\theta}{dt} \quad (2)$$

$$V_R = R_{RR} I_R + L_{RR} \frac{dI_R}{dt} + L_{RS}(\theta) \frac{dI_S}{dt} + \frac{\partial L_{RS}(\theta)}{\partial \theta} I_S \frac{d\theta}{dt} \quad (3)$$

In these equations,  $V_S$  represents the voltages at the stator terminals and  $V_R$  is the voltage at the rotor terminals which, in the case of the squirrel-cage motor, is equal to zero.  $R_S$  and  $R_R$  are the matrix representation of the stator and rotor resistances, respectively.

### 2.3 Simulation of the Three-Phase Induction Motor

Figure 2 shows the model of the three-phase induction motor resulting from the equations of voltages (1) and (2) projected on the  $d-q$  axes, and referred to the stator and the rotor respectively. It also shows the block that generates the voltages from the maximum voltage, the frequency and establishing a phase difference of  $120^\circ$  electric degrees between each phase. This model is implemented using Matlab/Simulink blocks that allows to simulate the electric motor in the following axes reference frames:

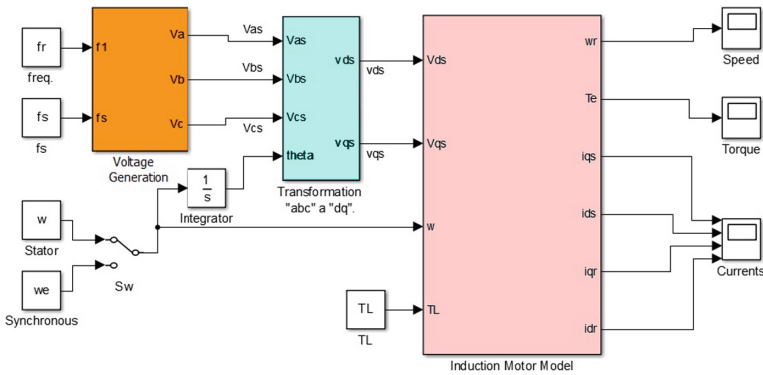


Fig. 2 Induction motor model.

- Reference frame fixed to the stator magnetic field (synchronism speed) ( $w = w_e$ ).
- Reference frame fixed to the rotor ( $w = w_R$ ).
- Reference frame fixed to the stator ( $w = 0$ ).

### 2.4 Simulation Results

The motor data used in the simulation are presented in Table 1.

Table 1. Induction motor parameters.

Parameter	Value	Parameter	Value
$P$	2 poles	$L_R$	144.6 [mH]
$R_S$	1.0 [ohm]	$L_M$	136.0 [mH]
$R_R$	1.3 [ohm]	$Jm$	0.022 [Kg m <sup>2</sup> ]
$L_S$	35.4 [mH]		

The resulting rotor speed of the electric motor model is shown in Fig. 3. It can be verified that the motor speed reached 377 rad/sec in more than 1.5 s.

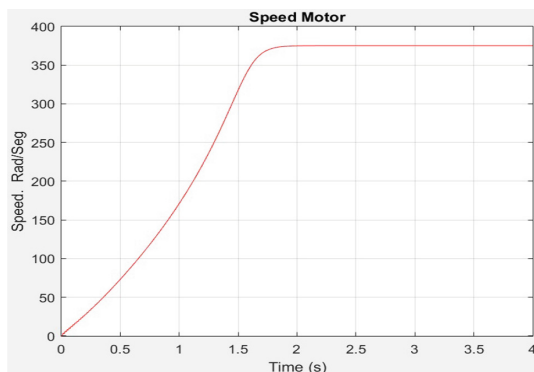


Fig. 3. Rotor speed at no load condition.

### 3 Vector Control of the Induction Motor

Vector control, also called Oriented Field Control (FOC), was proposed Hasse [3] and Blaschke [4]. This control strategy allows independent control of magnetic field flux and electromagnetic torque. This is achieved by reducing the non-linear model of the motor to a model referred to two orthogonal axes  $d$ - $q$ , which allows the induction motor to be modeled in the same way as a separate excitation DC motor [5]. Transforming from a three-phase system to a two-phase system, using space vectors and the transformation  $\alpha$ - $\beta$  to a system of axes  $d$ - $q$ . Allowing to control the magnetic flux by means of the regulation of the current component of the direct  $d$  axis, while the torque is controlled by regulating the current associated with the quadrature  $q$  axis.

Figure 4 shows the vector diagram, with the axes of coordinates  $\alpha$  and  $\beta$  stationary, and  $d$  and  $q$  the components of the vector axes, which are rotated by an angle  $\lambda$ .

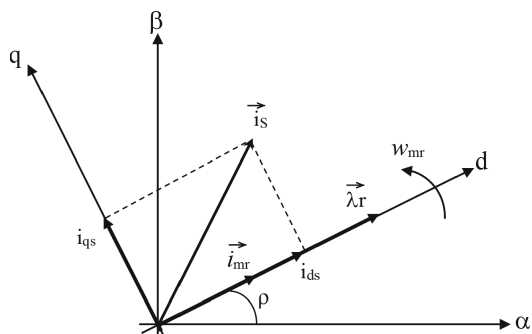


Fig. 4. Vector diagram of the axes  $d$ - $q$ .

The Field Oriented Controller (FOC) is responsible for positioning the magnetic flux vector directly on the axis  $d$ , so that the quadrature current is zero ( $i_{sq} = 0$ ). The magnetic flux must also be kept constant so that the electromagnetic torque is proportional to the variation of the current in the axis  $q$ . Thus the motor is controlled indirectly through the stator current ( $\vec{i}_s$ ). Through its two projections on the axes  $d$ - $q$  provide two components, real part ( $i_{sd}$ ) controls the magnetic flux and the imaginary part ( $i_{sq}$ ) controls the electromagnetic torque.

$$\vec{i}_s = i_{sd} + ji_{sq} \tag{4}$$

Where:

$i_{sd} \propto \lambda$ : Stator current in the  $d$ -axis component proportional to the magnetic flux.

$i_{sq} \propto T_e$ : Stator current in the  $q$ -axis component proportional to the torque.

$\vec{i}_{mR} = \vec{\lambda}_R / L_M$ : Magnetization current (magnetic flux producing).

### 3.1 Indirect Vector Control Method

The angle  $\lambda$  for the indirect vector control method is obtained through the slip calculation, applying Eq. 7. Figure 5 shows the general indirect scheme of vector control, the inputs of said system being the  $i_{sd}$  currents (reference values of magnetic flux) and  $i_{sq}$  (electromagnetic torque reference values).

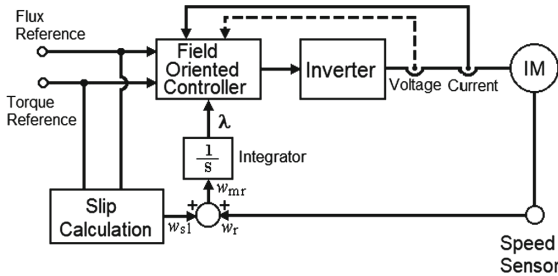


Fig. 5. Scheme of indirect vector control.

$$\lambda_r(s) = \frac{L_m}{1 + \tau_r s} i_{ds}(s) \tag{5}$$

$$w_{sl} = \frac{L_m i_{qs}}{\tau_r \cdot \lambda_r} \tag{6}$$

$$w_{sl} = \frac{L_m i_{qs}}{\tau_r \cdot \lambda_r} = \frac{1}{\tau_r} \frac{i_{qs}}{i_{ds}} (\tau_r s + 1) \tag{7}$$

## 4 Implementation of the Estimator Using Neural Networks

The artificial intelligence technique used to estimate the electric motor speed was the Artificial Neural Networks (ANN). This choice is due to the good results obtained using this technique in systems with nonlinear dynamics [6]. The type of neural network chosen is “*feedforward*” with automatic back-propagation and supervised learning [7].

The designed RNA has 6 inputs, which correspond to 3 voltage signals and 3 current signals, that is, a voltage sensor and a current sensor for each phase of the motor. Figure 6 shows the block diagram of the proposed estimator.

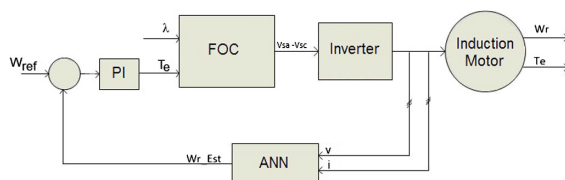


Fig. 6. Block diagram of the proposed estimator.

The implementation of the Artificial Neural Network to carry out the estimation of the speed will be executed in a Raspberry Pi development card, in conjunction with the Simulink/Matlab programming environment. In this environment, the necessary algorithms were developed, which were then turned over to the development card, from which they were executed autonomously. On the other hand, the signals delivered by the current and voltage sensors enter the development card by means of the GPIO pins, and are handled, within the Simulink environment, by means of the I/O blocks.

Figure 7 shows the blocks that conform the vector control loop to be used with the real electric motor.

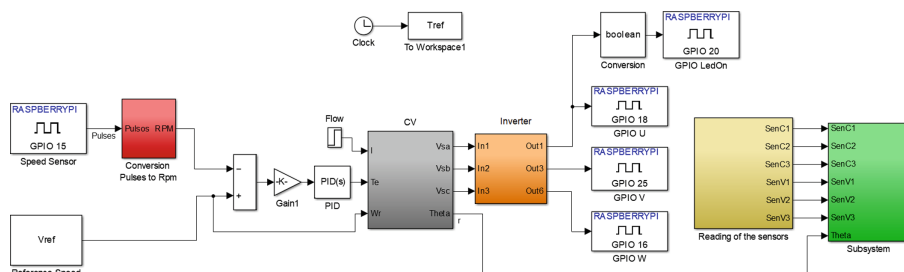


Fig. 7. Simulink model of the vector control for tests. (Color figure online)

GPIO blocks are used to send and/or receive data to/from the Raspberry Pi development card. In this case, the GPIO 15 block is the port through which the pulses delivered by the speed sensor are received. These input pulses are translated at a speed in rpm by means of a conversion block. Figure 7 shows the mention block highlighted in red.

The block corresponding to the inverter is highlighted in orange, and is responsible for receiving the signals from the vector control strategy and transforming them into PWM signals, which drive the power circuits, and which are sent to the development card by means of GPIO blocks 16, 18 and 25.

The reading of the voltage and current sensors was done using an Arduino development card, by means of the Analog/Digital converter. Thus, the signals from the sensors are converted into binary data and sent to the Raspberry Pi card.

Figures 8 and 9 show the voltage and current signals taken from the real electric motor. In this case, the signals obtained at three different speeds are shown: 200 rpm, 500 rpm and 800 rpm. These will later be used for the training of the neural network.

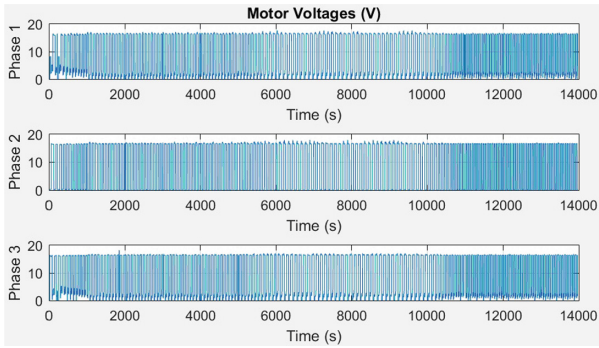


Fig. 8. Measured motor voltages per phase.

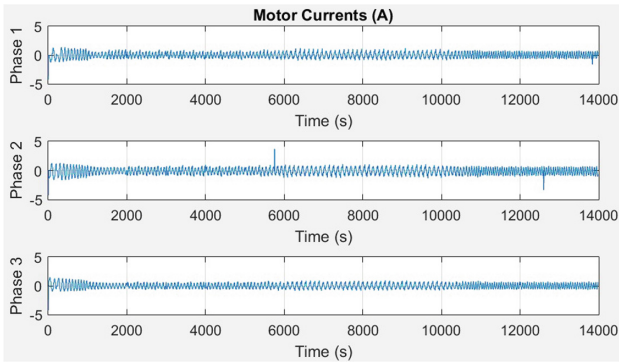


Fig. 9. Measured motor currents per phase.

The voltage and current data are sent to the Matlab Workspace, and are stored in the matrices  $V_S$  and  $I_S$  respectively. Subsequently, these matrices are concatenated in a single matrix called  $Inp$ .

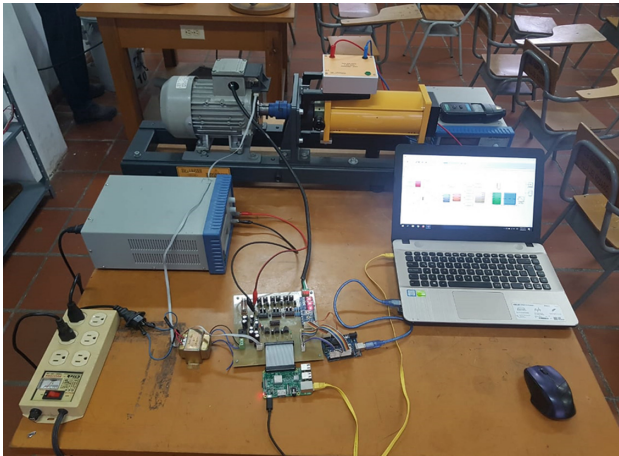
Once the training is done, we proceed to generate the block in Simulink that contains the trained RNA, using the *gensim* function.

## 5 System Validation

To implement the speed control strategy of the induction motor, the following components are used:

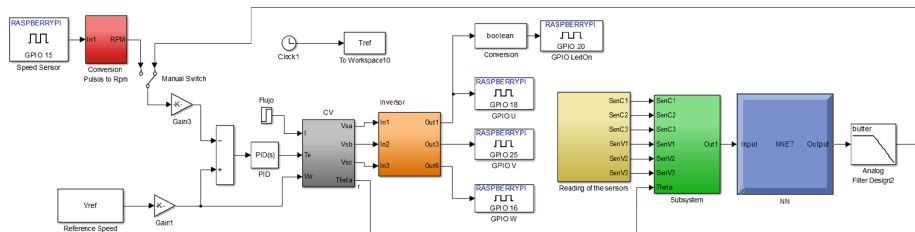
- Raspberry Pi Model 3B card.
- Electric power inverter.
- Current sensors.
- Voltage sensors.
- Arduino UNO card.
- Induction motor.

Figure 10 shows a photograph of the entire test bench used to carry out the experiments concerning the development of the project.



**Fig. 10.** Experimental setup.

The following figure shows the block diagram of the neuronal estimator, where the RNA block (blue block) is added, which is in charge of estimating the speed of the motor, depending on the voltages and currents of each phase of the machine. This estimated speed signal will be used to close the indirect control loop (Fig. 11).

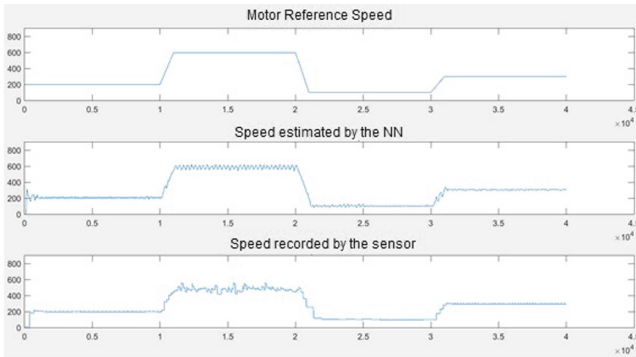


**Fig. 11.** Block diagram of the neuronal estimator. (Color figure online)



### 5.1 Estimator Results Using Neuronal Networks

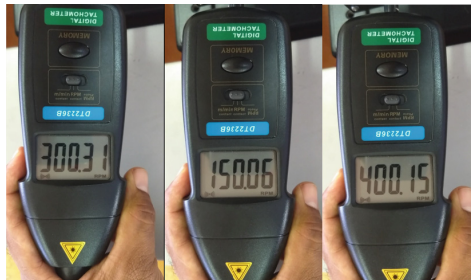
The results obtained from the implementation of the vectorial control loop. The tests made with different speeds and how the system behaves both in vacuum and under load are shown. The robustness of neural networks for Sensorless vector control is also evidenced when load is applied to the engine.



**Fig. 12.** Response of the speed estimation for 100, 200, 300 and 600 rpm.

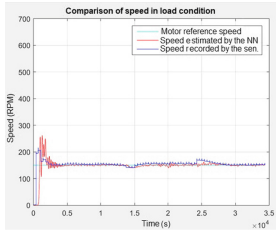
The above image shows the reference velocity (up), the velocity estimated by the ANN (center) and the velocity measured by the horseshoe sensor (down). It starts with a speed of 200 RPM, then changes speed to 600 RPM, then lowers the reference speed to 150 RPM and finally sets the speed to 300 RPM.

To validate that the motor speed is correct, a manual tachometer was used to corroborate the speed of the three-phase motor.

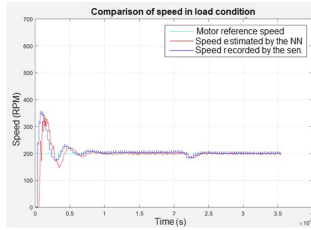


**Fig. 13.** Response of the speed estimation for 300, 150 and 400 rpm with the tachometer.

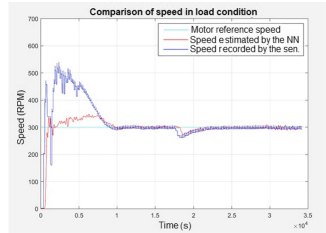
In the following images you can see the behavior of the estimation strategy, velocities obtained with load at 150, 200 and 300 rpm. The sequence was as follows: the system was started and at 15 s the load was applied, it is observed that the system is stabilized at the reference value; at 25 s the load was removed, it is observed that the speed of the system tends to increase, but the control strategy makes it arrive, again, at the reference value.



**Fig. 14.** Load at 150 rpm.  
(Color figure online)



**Fig. 15.** Load at 200 rpm.



**Fig. 16.** Load at 300 rpm.

Figure 14 shows the comparison of the speed in load condition. The red line is the control response applying the estimate with neural networks and the blue line is the real speed given by the sensor. It is observed that the estimated value tends to follow the real value after a few tenths of seconds showing a fairly good convergence.

From 1.5 s the vector control response is evaluated using the estimated speed. It is observed that the vector control responds adequately, i.e. both the estimation and the control are operating correctly. At 2.5 s the load is removed and the system responds again correctly. This indicates that the motor speed estimate is correct.

The behavior of the estimate at 200 rpm, for which it was not trained, shows the learning of the network and its satisfactory response as shown in Figs. 14 and 15 the robustness of the estimation strategy tends to decrease in the transient of the response, but instead continues to converge in the rest of the estimate, indicating that the response in its entirety is considered satisfactory.

## 6 Conclusions

It was demonstrated, by means of this work, that it is possible to close a control loop for a three-phase squirrel cage type induction machine, using, instead of a speed sensor, an ANN that estimates the motor speed using the currents and voltages of each of the phases of the machine. This is evidenced in the analysis of Figs. 13, 14 and 15, which show the comparison between the simulated system, the estimated ANN speed and the actual motor speed.

Efforts were concentrated on controlling the machine in an RPM range below 500. This was to demonstrate that the Sensorless vector control technique is effective even in this speed range. Again, Figs. 12, 13, 14, 15 and 16 show positive results in this scenario.

It was demonstrated that the set SIMULINK - Raspberry Pi 3B, is a solution, at hardware level, valid for the problem presented: to control a three-phase induction machine, at low speeds, using the Sensor-less vector control strategy.

It was demonstrated, through the work done, that it is possible to design and train an ANN that estimates the speed of an induction machine, using as training data

various patterns of current and voltage, characteristic of different speeds of rotation of the machine.

## References

1. João, C., Palma, P.: *Accionamentos Eletromecânicos de Velocidade Variável*. Fundação Calouste Gulbenkian, Lisboa (1999)
2. Barbi, I.: *Introdução ao Estudo do Motor de Indução*. Universidade Federal de Santa Catarina (1988)
3. Hasse, K.: *Zur Dynamic Drehzahleregelter Antriebe mit Strommaschinen*, techn, Hochschule Darmstadt, Dissertation (1969)
4. Blaschke, F.: The principle of field orientation as applied to the new “Transvektor” close-loop control system for rotating-field machines. *Siemens Rev.* **39**(5), 517–525 (1972)
5. Kakhim, A.H., Rezak, M.J.A., O’Kelly, D.: A PWM inverter using on-line control algorithms. In: *UPEC 1991*, pp. 333–336 (1991)
6. Pyne, M., Chatterjee, A., Dasgupta, S.: Speed estimation of three phase induction motor using artificial neural network. *Int. J. Energy Power Eng.* **3**(2), 52–56 (2014)
7. Ponce Cruz, P.: *Inteligencia artificial con aplicaciones a la ingeniería*. Alfaomega, México (2010)
8. González, J., Azevedo, M., Pacheco, A.: Control vectorial del Motor de Inducción para el Control de Velocidad del Rotor por cambio de Frecuencia, II Congreso Venezolano de Ingeniería Eléctrica, Mérida (2000)
9. González, J., Silveira, M., Pacheco, J.: Comparación de la red neuronal y el filtro de Kalman en la estimación de velocidad del motor de inducción (2004)
10. Gallo, M., Duran, C.: Integrated system approach for the automatic speech recognition using linear predict coding and neural networks. In: *IEEE Electronics, Robotics and Automotive Mechanics Conference, CERMA 2007* (2007)
11. Quintero, C.M., Rodríguez, J.L.D., García, A.P.: Aplicación de redes neuronales al control de velocidad en motores de corriente alterna. *Revista Colombiana de Tecnologías de Avanzada* **2**(20), 113–118 (2012)