# Optimum Fuzzy Logic Controller Apply to Induction Motor Control

JORGE L. DIAZ R, ALDO PARDO G. *Member, IEEE,* TANIA L. ACEVEDO G.
University of Pamplona, Pamplona
jdiazcu@gmail.com, apardo13@hotmail.com, tl_acevedo@hotmail.com

*Abstract:* - This work deals with the design of an optimized Fuzzy Logic Controller using genetic algorithm for the speed of an induction controlled motor by a variable frequency converter. For this reason a mathematically structured genetic-fuzzy algorithm was developed and implemented in a microprocessor, Basic X-24. Finally, to display the output and to obtain the motor transient response, we used a data acquisition card (PCI 6023E) and a software developed in LabView®.

*Key-Words:* - Induction motor drives, Fuzzy control, Optimization methods, Genetic algorithm, Data acquisition.

## 1. INTRODUCTION

AT present, the use of artificial intelligence in the field of power electronics and speed control systems among others has increased in the last two decades [1], [2]. The Knowledge Base (KB) is the Fuzzy System (FS) component comprising the expert knowledge knows about the problem. So is the only component of the FS depending on the concrete application and it makes the accuracy of the FS depends directly on its composition. The KB is comprised of two components, a Data Base (DB), containing the definitions of fuzzy rules linguistic labels, that is, the membership functions of the fuzzy sets, and a Rule Base (RB), constituted by the collection of fuzzy rules representing the expert knowledge [3].

This paper addresses the application of optimized fuzzy logic control as a technique to be applied to electrical machines and drives, specifically, the speed control of induction motors. Our main goal is to improve the speed characteristics and their recovery when load change in the motor, it is also called load disturbances, and is being analyzed. A speed controller with genetic algorithm and fuzzy logic techniques based on the mathematical models of the motor and the drive ware developed, along with their simulations.

## 2. DESIGN OF A MAMDANI FUZZY LOGIC CONTROLLER

Inspired by the work of Zadeh [3], Mamdani [4] and their collaborators [5] combined the ideas of the rules based on systems using fuzzy parameters to build a controller based on the reasoning of a human operator [6], [16]., [17].

To implement this methodology, it has to mimic the designer logic on the partition of the input space and the approach within the design of control strategies involving expert knowledge.

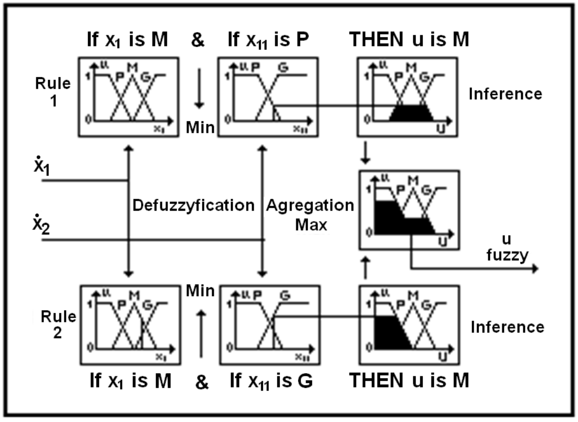


Fig. 1. Inference algorithm with defuzzification.

The fuzzy control systems can be used for this purpose. The direct relationship of the control scenarios and its input - output relations regarding the fuzzy statements (IF – THEN) it can be used to design the Fuzzy Logic Controller (FLC).

### A *The input sets*

For the formation of the controller inputs, the simulations results should be taken into account, which verifies the achievement of the best results with a small length of the discourse universes; therefore, the values of -0.5 to 0.5 were taken for the two inputs, the error and the error rate. (See figures 2 and 3).

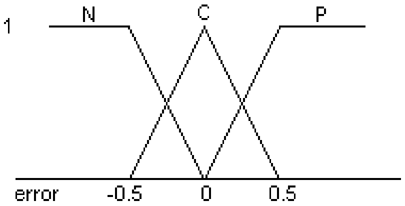


Fig. 2. First input set: Error.

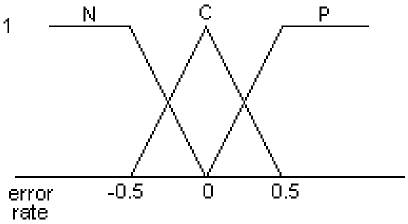


Fig. 3. Second input set: Error rate.

### *B* ***The output reference set***

The first step is the discretization of the output reference set according to the voltage level of the control device.

$$Yd = \left[\,0\ 0.25\ 0.5\ 0.75\ 1\ 1.25\ 1.5\ 1.75\ 2\ 2.25\ \cdots\right.$$
$$\left.\cdots 2.5\ 2.75\ 3\ 3.25\ 3.5\ 3.75\ 4\ 4.25\ 4.5\ 4.75\ 5\right]$$

Taking into account the simulation results for the distribution of the output fuzzy set raises the required voltage of the following sets of fuzzy output.

### *C* ***The output fuzzy sets***

$$Ub = \left[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0.5\ 0\ 0\ 0\right]$$
$$Un = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0.5\ 1\ 0.5\ 0\right]$$
$$Ua = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0.5\ 1\right]$$

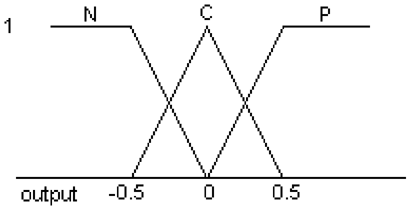


Fig. 4. The output set: Output

### ***D*** ***The rules' base***

Once the input and output sets were stated, the next step is to declare the 9 rules' base that makes up the fuzzy logic controller.

The $D_i$ sets describe the output activation sets under the designed rules for each preceding statement.

$$D_1, D_2, D_4 = \left[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0.5\ 0\ 0\ 0\right]$$
$$D_3, D_5, D_5 = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0.5\ 1\ 0.5\ 0\right]$$
$$D_6, D_8, D_9 = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0.5\ 1\right]$$

### ***E*** ***Triggering rules for activation***

The next step of the inference algorithm is to calculate the triggering degree of each rule.

Where $\tau_i$ denote the triggering degree from the *i*-th rule and is given by:

$$\tau_i = [\mu A_1(x_1) \wedge \mu B_{i1}(x_1)] \tag{1}$$

### ***F*** ***Evaluation of the individual outputs***

The following step is to obtain the fuzzy output sets inferred by every rule. The degrees of membership of output fuzzy sets are represented as follows:

$$F_1(y) = \tau_1 \cap D_1(y) \tag{2}$$

### ***G*** ***Aggregation of the individual outputs***

The last step of inference algorithm is to obtain the union of the individual output sets; this step depends on the number of outputs' activations according to the inputs:

$$\begin{aligned} F(y) = F_1(y) \cup F_2(y) \cup F_3(y) \cup F_4(y) \cup F_5(y) \cdots \\ \cdots \cup F_6(y) \cup F_7(y) \cup F_8(y) \cup F_9(y) \end{aligned} \tag{3}$$

### ***H*** ***Defuzzification***

To get the $\Delta u(k)$ value the centroid method was applied.

$$\Delta u(k)^* = y^* = \frac{\sum_{i=1}^{9} F(y) \times Y_d}{\sum_{i=1}^{9} F(y)} \tag{4}$$

## 3. OPTIMUM FUZZY LOGIC CONTROLLER USING GENETIC ALGORITHM

Genetic Algorithm (GA) is a powerful problem solving tool [7] which attempts to mimic the natural selection process and evolution in order to find the optimal solution to a given problem, it is a mathematical method which is motive by the concept of survival of the fittest in biological evolution. The first step of optimization of fuzzy logic control functions is the creation of population of fuzzy membership functions [8] . The Genetic algorithms (GA) have demonstrated to be a powerful tool for automating the definition of the KB since adaptive control, learning and self-organization can be considered in a lot of cases as optimization or search process. The fuzzy systems making use of GA in their design process are called generically GFSs.

These advantages have extended the use of Gas in the development of a wide range of approaches for designing FSs in the last years. It is possible to distinguish three different groups of genetic FS design process according to the KB components included in the learning process. These ones are the following:

- Genetic definition of the Fuzzy System Data Base [9], [10], [11].

- Genetic derivation of the Fuzzy System Rule Base ([12], [8].
- Genetic learning of the Fuzzy System Knowledge Base [12], [13], [14], [5].

### *A* ***Background of theme.***

In this paper it is shown how a genetic algorithm can be used in order to optimize a fuzzy system which is used for the speed of an induction controlled motor by a variable frequency converter. This research will work on the controller designed by the expert system, to get a better set of rules of inference with the genetic algorithm; this genetic process is shown in the flow diagram of Figure 5.

1. *Encoding scheme:* GAs work with a population of chromosomes, each of which can be decoded into a solution of the problem. Encoding scheme in genetic algorithm is the basis of its development, which directly affects the construction of genetic operators and performance of genetic algorithm.

2. *Initial population:* An initial population is created from a random selection of solutions.

3. *Fitness function:* During the search procedure, each individual is evaluated using the fitness function. Our objective is to maximize battle effectiveness.

4. *Selection:* For selection, two individuals are randomly chosen from the population and they form a couple for crossover. Selection can be based on different probability distributions, such as uniform distribution or a random selection from a population where each individual is assigned a weight dependent on its fitness, so that the best individual has the greatest probability to be chosen. We order the individuals of the population according to their fitness values. The selection probability for the individuals is a linear distribution.

5. *Crossover:* The crossover operator generates new chromosomes. Crossover is usually applied to selected pairs of parents with a probability equal to a given crossover rate. In our algorithms, the idea behind a crossover operation is as follows: it takes as input 2 individuals, selects a random point, and exchanges the column behind this point as a whole. It avoids effectively unfeasible solution during crossover.

6. *Mutation:* The mutation operation modifies an individual. In defining the mutation operator, we take into account the domain type of an attribute. We consider an important situation: The individual must be a feasible solution.

7. *End criterion* If, the termination criterion (statistical or temporal) is not satisfied, then a return is made to the third step; otherwise, the algorithm is terminated. The criterion is usually a sufficiently good fitness, or a maximum number of iterations, or the global best fitness is steady-going within determinate iterations.

### *B* ***Process of optimization of Fuzzy Controller making use of GA.***

The genetic process starts with the creation of a population of individuals (created at random), each containing of values resulting from a fuzzy controller in particular. To form the rules basis of a fuzzy controller, it requires the values of the background, the type of connection to this case "and" and the given weight to each rule is 1.

This fuzzy controller is not intelligent by itself, even needs a little help from an expert to be able to develop it. The expert in this case is not helpful when planning the rules, only in defining the universe of discourse of each entry and exit of the driver and type of membership that was used.

Finally, along with the foundations of rules, the ranges of the input variables output data of the membership functions: the Role of the system has a file as output FIS (Fuzzy inference System), which corresponds to a structure where they have All data inference fuzzy controller. This structure FIS is used in the simulation by the control diffuse scheme.

After, the simulation was obtained the value of the index of performance, which is sent to the genetic algorithm, to continue with the process. For clarity, the entire process is presented in Figure 5

### *C* ***Optimized Fuzzy Logic Controller implementation***

For the optimized controller implementation the BasicX-24 microprocessor (www.basicx.com) was used which is a system based on a chip, combined with a software development environment compatible with the Windows platform [14], [15].

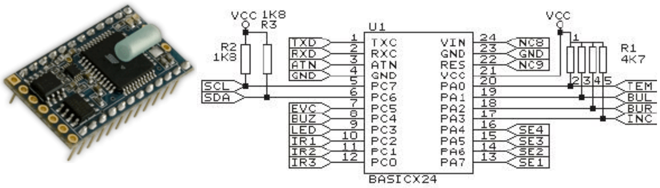


Fig. 5. BasicX-24 Microprocessor

The programming used was a Basic version, which has specific commands for this microcontroller and thus, obtains the final control actions. The microprocessor output has to be amplified to a maximum of 10 V and then it is fed into the power converter.

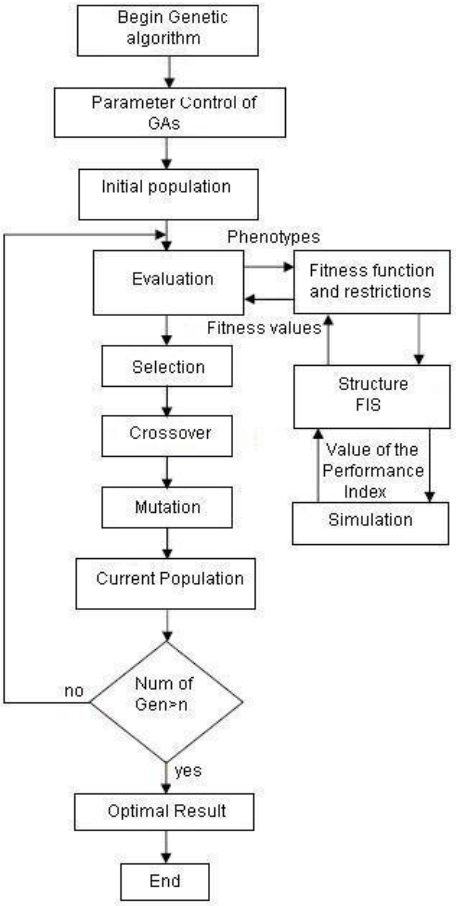


Fig.6. Process flow genetic diagram.

## 4. RESULTS AND DISCUSSION

The tests were implemented in a facility at the Electrical Machines' laboratory. This facility consists of a three-phase induction motor ($P_{out}$ = 300W), a DC generator (that serves as load and speed sensor at the same time), and a digital frequency converter (a didactic one), all of them of HPS System Technik. Additionally, two power sources were used: a direct current for the field of the DC generator and a three-phase alternating current as the power supply for the motor control [13], [14], [15].

Fig. 7. Experimental setup

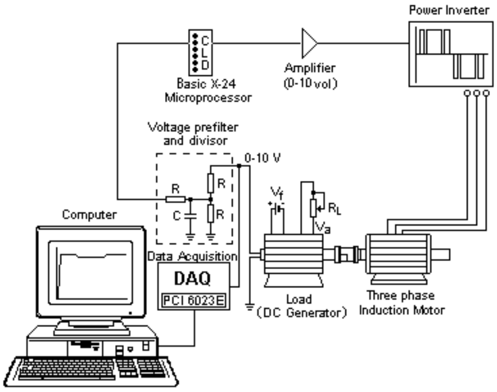
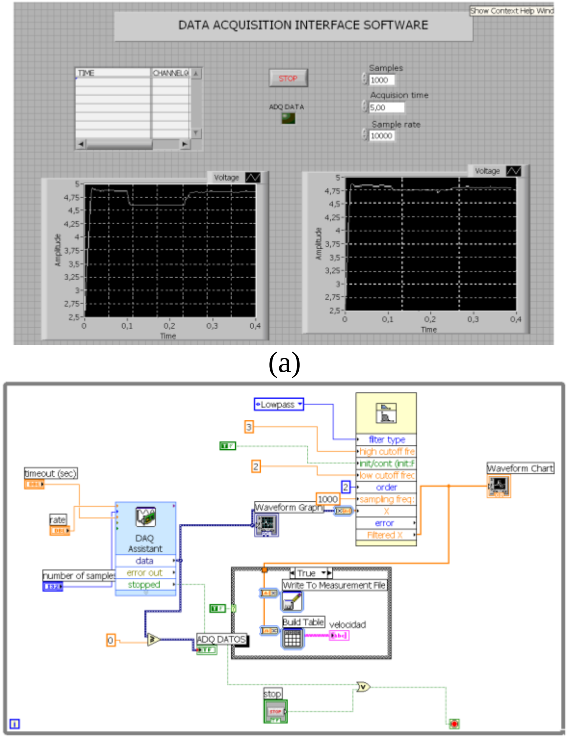


Fig. 8. Control system diagram

The experiment was conducted energizing the power inverter and when the motor stabilizes its speed, a sudden load (step) change was made. First, the load change reaches the 100% of the rated load, and later for the 50% of the rated load. These values were selected according to a very demanding condition that could at any time affect the motor; thus, the most demanding conditions have been considered. The sudden changes are considered here as any 'worst case scenario'. The motor output variable (the speed) was sensed by the DC generator. (See figures 7 and 8).



Samples 1000

Acquision time 5,00

Sample rate 10000

(a)

(b)

Fig. 9. Data acquisition interface with LabView®. (a) Instruments Panel (b) Connection diagram

A low-pass 2nd order Butterworth digital filter was calculated and programmed with LabView®, with a maximum frequency of 3 Hz and a minimum of 2 Hz. The use of this filter was essential to eliminate the noise, mainly due to electromagnetic emissions of the electrical machinery used. This amount of noise is induced and contaminates the speed signal taken from the motor. The speed signal is a slow variable and can be regarded as a DC value.

Once the motor is stabilized after a second, (3000 samples) the load is set and the speed is capture.

The *x*-axis shown in figure 10 represents samples (indirect measurement of time), as much as 3000 samples per second, (this takes about 4 seconds). The *y*-axis represents the motor speed that is about 1450 rpm and takes the voltage and the equivalence that is 10 V, equivalent to 3000 rpm.

Figure 11 shows the motor speed response at 100% of rated load, but unlike the previous response, with the system in closed loop, the load has very small influence on the motor speed; with the controller action the speed is not too drastically disturbed

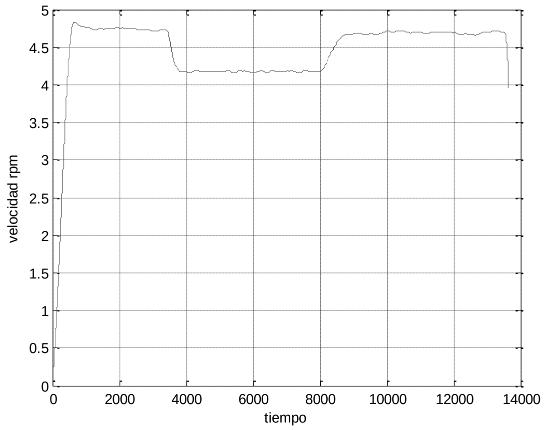


Fig. 10. Open loop response with a 100% load change

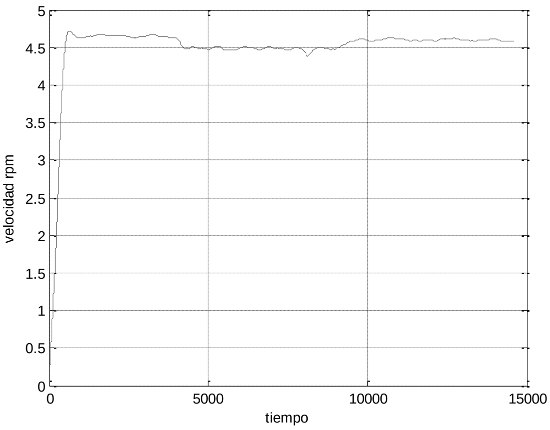


Fig. 11. Closed loop response with a 100% load change

The same experiment was carried out again but now with 50% of its rated load and the speed results were captured.

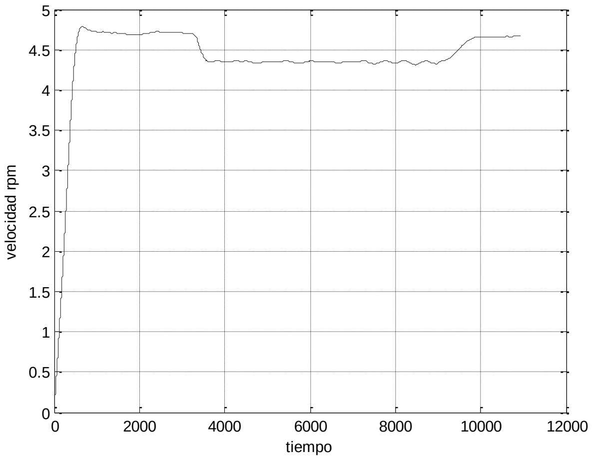


Fig. 12. Open loop response with a 50% load change

Figure 12 represents the output speed of the induction motor in open loop with 50% of the rated load; it appears that the speed reduction is much lower than the previous one with the application of 100 % of its rated load.

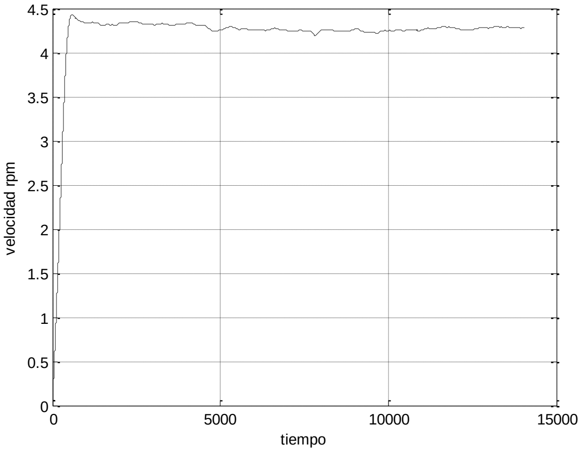


Fig. 13. Closed loop response with a 50% load change

In the case of a closed loop (see figure 13) the optimized fuzzy logic controller quickly compensates the load changes so that the system is virtually immune to any disturbance; it verifies the functionality of the designed controller as well as its good operation for different loads.

## 5. CONCLUSIONS

Intelligent control techniques such as, Optimized Fuzzy Logic Controller, are sufficiently robust and valid, starting with its ability to deal with disturbance in the control of various processes; in particular, with the induction motors. This was obtained in the experimental results.

As a result of the real time implementation, the captured samples taken during the experiment showed that the speed response of the induction with the optimized fuzzy logic controller dealing with sudden increase and decrease of load changes up to 50% and 100% of its rated load, are quickly compensated by the controller so that the system is virtually immune to disturbances. Thus, the functionality of the designed controller and its good performance to eliminate disturbances is verified.

The selection of rules meets the expected results as shown in the charts and this also shows that it is possible to create a robust control algorithm that presents very good response characteristics for motor speed control using power converters. Finally, it demonstrates with this experiment the effectiveness of the use of Optimized Fuzzy Logic Controller and its feasibility of its application in the field of electric motor drives.

## *References*

[1] T. Ying-Yu; H. Lien-Hsun, and O. Rong-Shyang, “Fuzzy Control of a Closed-loop Regulated PWM Inverter under Large Load Variations”. Proceedings of the IECON '93, 1993.

[2] W. Cheng, and Z. Cheng, “A Fuzzy Speed Controller of Induction Motors”. IEEE Trans. Chengdu, China, August, 2004.

[3] L. A. Zadeh. “Fuzzy sets”. Inform. Contr., Vol. 8, 1965, pp. 338-353.

[4] E. H. Mamdani. “Application of simply algorithm for control of a simple dynamic plant”. Proc. IEE, Vol. 121, 1974, pp. 1585-1588.

[5] J. Aguilar y E. F. Rivas. *Introducción a las técnicas de computación Inteligente.* Universidad de Los Andes, Venezuela, 2001.

[6] Coley D. An Introduction to Genetic Algorithms for Scientists and Engineers (WSP, 1999)(ISBN 9810236026)(T)(244s).

[7] Karr C., 1991. Genetics algorithms for fuzzy controllers. AI Expert T. L. pages 26-33.

[8] Herrera F., Lozano M. & Verdegay J. L., 1995a. A Learning process for fuzzy control rule using genetic algorithms. Technical Report DECSAI-95108, University of Granada, Department of Computer Science and Artificial Intelligence.

[9] Cooper M. G. & Vidal J. J., 1993. Genetic design of fuzzy logic controllers. In Proc. Second International Conference on Fuzzy Theory and Technology (FTT’93), Durham.

[10] Leitch D. & Probert P., 1994. Context depending coding in genetic algorithms for the sensing of fuzzy systems. In Proc. IEEE/Nagoya University WWW on Fuzzy Logic and Neural Networks/Genetic Algorithms Nagoya.

[11] T. L. Acevedo G. Diseño e Implementación de un controlador Lógico Difuso aplicado a un Motor de Inducción. Tesis de Maestría, Facultad de Ingenierías y Arquitectura, Universidad de Pamplona, Pamplona, Colombia, 2008.

[12] A. Pardo G. y J. L. Díaz R. Aplicaciones de los convertidores de frecuencia, Estrategias PWM, Editorial Java E. U., Colombia, 2004.

[13] C. A. Vera R. Estudio para la Obtención de los Procesos Transitorios en un motor de inducción. Tesis de Maestría. Universidad de Pamplona, Pamplona, Colombia, 2007.

[14] C. A. Vera R., C. M. Duran A. y J. L. Díaz R. “Sistema de adquisición de datos con Matlab - Simulink usando la tarjeta PCI 6023E de National Instruments”. Revista Colombiana de Tecnologías de Avanzada, Colombia, Vol. 2, No. 12, Agosto, 2008.

[15] Acevedo G., J. L. Díaz R. y A. Pardo G. “Control V/Hz en lazo cerrado de una máquina de inducción utilizando regulador por lógica difusa”. Revista Colombiana de Tecnologías de Avanzada, Colombia, Vol. 2, No. 12, Agosto, 2008.

[16] Constantin Volosenc, Control of Electrical Drives Based on Fuzzy Logic, WSEAS TRANSACTIONS on SYSTEMS and CONTROL, ISSN: 1991-8763 Issue 9, Volume 3, September 2008.

[17] Constantin Volosenc, Fuzzy Control of Electrical Drives, WSEAS International Conference on POWER SYSTEMS, ISSN: 1790-5117, Cantabria, Spain, 2008.