PID Controller adapted by a fuzzy inference system optimized by Genetic algorithm for Induction machine control

Abdeldjebar HAZZAB[1], Ismaïl Khalil BOUSSERHANE[1], Mokhtar KAMLI[2] and Mustapha RAHLI[2]
[1] University center of Bechar B.P 417 BECHAR 08000 ALGERIA
Tél. : 0021349815581 (91) Fax : 0021349815244
[2] University of Sciences and Technology of Oran, ORAN (31000) ALGERIA.

Abstract - **In this paper we present the control of an induction machine alimented by a PWM three-phase inverter using a controller with PID structure adapted by a fuzzy inference system (adaptive FLC-PI). However the major disadvantage of the fuzzy logic control is the lack of design techniques, for this purpose we propose an optimization technique of the fuzzy-adapter parameters (adaptive FLC-PI-GA). This technique have presented a good performances compared to the controller which have parameters chosen by the human operator (adaptive FLC-PI).**

Key words – fuzzy logic, genetic algorithm, PID controller, adaptation, optimization, vector control.

I. INTRODUCTION

Nowadays, like a consequence of the important progress in the power electronics and of micro-computing, the control of the AC electric machines known a considerable rise and a possibility of the real time implantation applications. The Induction machine (IM) known by its robustness, cost, reliability and effectiveness is the subject of several research. However, it is traditionally, for a long time, used in industrial applications that do not require high performances, this because of its high non-linearity and the presence of the coupling between the stator states and the rotor states. On the other hand, the direct current (D.C) machine was largely used in the field of the variable speed applications, where torque and are naturally decoupled and can be controlled independently by the torque producing current and the flux producing current. Since Blashke and Hasse [1,2] have developed the new technique known as vector control, the use of the induction machine becomes more and more(increasingly) frequent. This control can provide the same performance as achieved from a separately excited DC machine.

The controllers most often used in the industrial applications are the PID-type controllers because of their simple structures and good performances in a wide range of operating conditions [3]. In the literature, the PID controllers can be divided into two parts:

In the first part, the controller parameters are fixed during control operation. These parameters are selected in an optimal way by known methods such as the poles imposition, Zeigler and Nichols... etc. The PID controllers of this part are simple but they have a disadvantage that they are linear and cannot control systems with changing parameters and have a high non-linearity [4].

In the second part, the controllers have an identical structure to PID controllers but their parameters are adapted on-line parameters estimation of the process. These controllers are known as adaptive PID controllers.

In control by fuzzy logic, the linguistic description of the expertise human is presented in the forms of fuzzy rules or relations for controlling the system. The controllers based on fuzzy logic (FLC) can be considered non-linear PID where their parameters are determined on-line based on the error and its derivative. Controllers FLC need much information to compensate for non-linearity when the operation conditions change [5]. Moreover, if the number of the inputs of the FLC increases the dimension of the rules base increases.

To overcome the disadvantages of PID controllers and FLC, we propose in this paper a combination between the two types of controllers. PID parameters controller can be adjusted by an adaptive mechanism based on a fuzzy inference (adaptive FLC-PI) for the speed control of a IM alimented by an PWM inverter.

However, the major drawback of fuzzy control is the lack of design techniques[6]. Most of the fuzzy rules are human knowledge oriented and hence rules will deviate from person to person in spite of the same performance of the system. The selection of suitable fuzzy rules, membership functions and their definitions along the universe of discourse always involve a painstaking trial-and-error process. Ga most known and is most largely employed in the technique of global research with a capacity to explore and exploit a given operation space using the measurement of the available performance [7, 8]. Recently of many applications combining the fuzzy concepts and Ga appeared, particularly, the use of Ga for the fuzzy logic systems control design. These approaches thus approaches are called *genetic-fuzzy system*. In this way, we propose a technique to optimize the parameters of fuzzy adapter of controller PI; the controller resulting from this combination is known on the name: adaptive FLC-PI-GA in order to apply it to the speed control of the induction machine.

II INDUCTION MACHINE MODEL

Fig. 1 bellow gives three different reference frames: stator reference frame ($\alpha - \beta$), rotor reference frame (*D-Q*) and arbitrary reference frame (*d-q*).

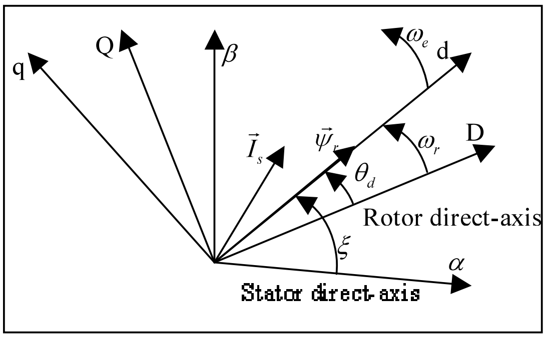


Fig. 1: Reference frames and space vector representation

The induction motor mathematical model, in space vector notation [6,7,8], established in *d-q* co-ordinate system rotating at speed ω_e is given by the following equations.

$$\vec{V}_s = R_s\vec{i}_s + \frac{d\vec{\psi}_s}{dt} + j\omega_e\vec{\psi}_s \quad (1)$$

$$0 = R_r\vec{i}_r + \frac{d\vec{\psi}_r}{dt} + j(\omega_e - \omega_r)\vec{\psi}_r \quad (2)$$

The stator and the rotor fluxes are given by:

$$\vec{\psi}_s = L_s\vec{i}_s + L_m\vec{i}_r \quad (3)$$

$$\vec{\psi}_r = L_m\vec{i}_s + L_r\vec{i}_r \quad (4)$$

The produced electromagnetic torque is given by:

$$T_e = \frac{3pL_m}{2L_r}(\vec{\psi}_r \otimes \vec{i}_s) \quad (5)$$

Using the *d-q* co-ordinate system, as illustrated in fig. 1, and separating the machine variables state vectors into their real and imaginary parts, the well-known induction motor model expressed in terms of the state variables is obtained from (1)-(5).

This model is given by equation (6):

$$\begin{cases} \dfrac{di_{sd}}{dt} = -\left(\dfrac{R_s}{\sigma L_s} + \dfrac{1-\sigma}{\sigma\tau_r}\right)i_{sd} + \omega_e i_{sq} + \dfrac{L_m}{\sigma L_s L_r \tau_r}\psi_{rd} + \dfrac{L_m\omega_r}{\sigma L_s L_r}\psi_{rq} + \dfrac{1}{\sigma L_s}V_{sd} \\ \dfrac{di_{sq}}{dt} = -\omega_e I_{sd} - \left(\dfrac{R_s}{\sigma L_s} + \dfrac{1-\sigma}{\sigma\tau_r}\right)I_{sq} - \dfrac{L_m\omega_r}{\sigma L_s L_r}\psi_{rd} + \dfrac{L_m}{\sigma L_s L_r \tau_r}\psi_{rq} + \dfrac{1}{\sigma L_s}V_{sq} \\ \dfrac{d\psi_{rd}}{dt} = \dfrac{L_m}{\tau_r}i_{sd} - \dfrac{1}{\tau_r}\psi_{rd} + (\omega_e - \omega_r)\psi_{rq} \\ \dfrac{d\psi_{rq}}{dt} = \dfrac{L_m}{\tau_r}i_{sq} - (\omega_e - \omega_r)\psi_{rd} - \dfrac{1}{\tau_r}\psi_{rq} \\ \dfrac{d\omega_r}{dt} = \dfrac{p^2 L_m}{JL_r}\left(i_{sq}\psi_{rd} - i_{sd}\psi_{rq}\right) - \dfrac{f_c}{J}\omega_r - \dfrac{p}{J}T_l \end{cases} \quad (6)$$

Where σ is the coefficient of dispersion and is given by (7):

$$\sigma = 1 - \frac{L_m^2}{L_s L_r} \quad (7)$$

L_s, L_r, L_m stator, rotor and mutual inductances;

R_s, R_r stator and rotor resistances;

ω_e, ω_r electrical and rotor angular frequency;

ω_{sl} slip frequency $(\omega_e - \omega_r)$;

τ_r rotor time constant (L_r / R_r);

p pole pairs

III. INDIRECT FIELD-ORIENTED CONTROL OF AN INDUCTION MOTOR

The main objective of the vector control of induction motors is, as in DC machines, to independently control the torque and the flux; this is done by using a *d-q* rotating reference frame synchronously with the rotor flux space vector [6,7] as shown in fig. 1, the *d* axis is aligned with the rotor flux space vector. Under this condition we have:

$\psi_{rq}^* = 0$ and $\psi_{rd}^* = \psi_r^*$

For the ideal state decoupling the torque equation become analogous to the dc machine as follows:

$$T_e = \frac{3}{2}\frac{P \cdot L_m \cdot \psi_r}{L_r} \quad (8)$$

And the slip frequency can be given as follow:

$$\omega_{sl} = \frac{1}{\tau_r}\frac{i_{sq}^*}{i_{sd}^*} \quad (9)$$

Consequently, the dynamic equations (6) yield:

$$\begin{cases} \dfrac{di_{sd}}{dt} = -\left(\dfrac{R_s}{\sigma L_s} + \dfrac{1-\sigma}{\sigma\tau_r}\right)i_{sd} + \omega_e i_{sq} + \dfrac{L_m}{\sigma L_s L_r \tau_r}\psi_{rd} + \dfrac{1}{\sigma L_s}V_{sd} \\ \dfrac{di_{sq}}{dt} = -\omega_e i_{sd} - \left(\dfrac{R_s}{\sigma L_s} + \dfrac{1-\sigma}{\sigma\tau_r}\right)i_{sq} - \dfrac{L_m\omega_r}{\sigma L_s L_r}\psi_{rd} + \dfrac{1}{\sigma L_s}V_{sq} \\ \dfrac{d\psi_r}{dt} = \dfrac{L_m}{\tau_r}i_{sd} - \dfrac{1}{\tau_r}\psi_{rd} \\ \dfrac{d\omega_r}{dt} = \dfrac{3}{2}\dfrac{p^2 L_m}{JL_r}i_{sq}\psi_{rd}^* - \dfrac{f_c}{J}\omega_r - \dfrac{p}{J}T_l \end{cases} \quad (10)$$

Fig. 2 shows the implemented diagram of an induction motor indirect field-oriented control (IFOC)[4,6]:

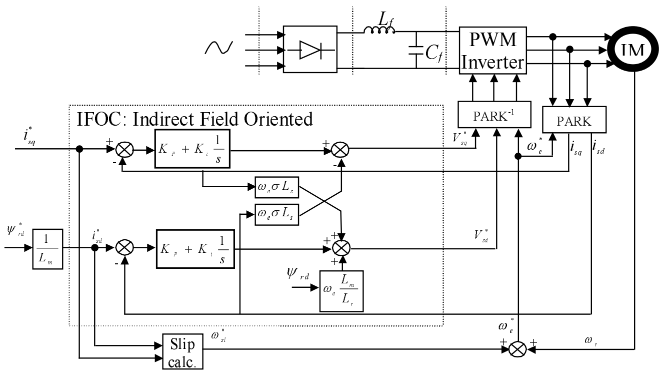


Fig. 2: bloc diagram of IFOC for an induction motor.

III. THE SPEED CONTROL OF THE IM BY AN ADAPTIVE CONTROLLER FLC-PI

To overcome the disadvantages of PID controllers and FLC, we propose in this paper a combination between the two types of controllers. PID parameters controller can be adjusted by an adaptive mechanism based on a fuzzy inference (adaptive FLC-PI). In what follows we show the method of combination between these two types of controllers.

A. Gain Adjustment by fuzzy logic

The adjustment of the gains is a technique which acts on the parameters of PI controller (k_p , k_i) to tune them during control of the system. This makes PI controller adapted to the nonlinear systems [5]. The diagram of this technique is illustrated in fig. 3. The fuzzy adapter adjust the PI parameters and generates new parameters to him, so that it adapts to all the operating conditions based on the error and its derivative.

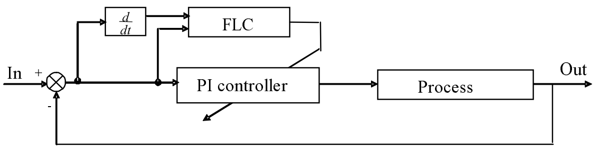


Fig. 3: PI control system with fuzzy gain adapter

B. Description of the fuzzy adapter

The parameters of PI controller used in the direct chain k and k_p are normalized into the range [0,1], by using the following linear transformations [3]:

$$k_p^{'} = \left(k_p - k_{p\min}\right) \Big/ \left(k_{p\max} - k_{p\min}\right) \tag{11}$$

$$k_i^{'} = \left(k_i - k_{i\min}\right) \Big/ \left(k_{i\max} - k_{i\min}\right) \tag{12}$$

The input of the fuzzy adapter are: The error e, the derivative of error Δe the outputs are: the normalized value of the proportional action (k_p) and the normalized value of the integral action (k_i). The fuzzy subsets of the input variables are defined as follows: NB: Negative Big, NM: Negative Medium, NS: Negative Small, ZE: Zero, PS: Positive Small, PM: Positive Medium, PB: Positive Big. The fuzzy subsets of the output variables are: B: Big, S: Small.

The membership functions for the inputs e and Δe are defined in the range [-1,1] (Fig 4), and for the outputs are defined in the interval [0,1] (Fig. 5).

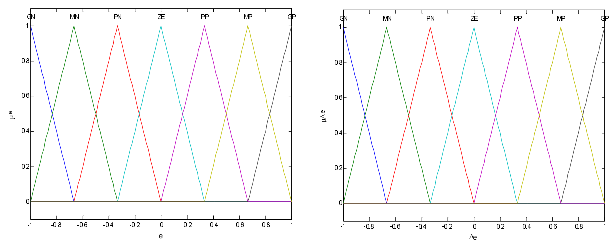


Fig. 4 : Membership functions e et Δe .

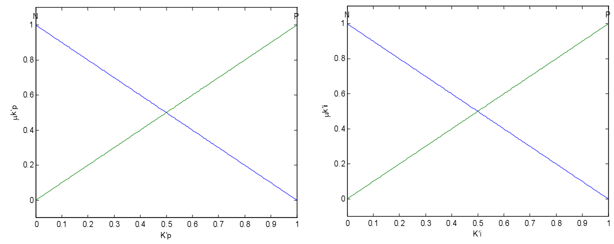


Fig. 5 : Membership functions $k_p^{'}$ et $k_i^{'}$

The rule base for computing k and k_i are shown in tables I and II respectively.

Once the values of k and k_i are obtained, the new parameters of PI controller is calculated by the following equations:

$$k_p = \left(k_{p\max} - k_{p\min}\right).k_p^{'} + k_{p\min} \tag{13}$$

$$k_i = \left(k_{i\max} - k_{i\min}\right).k_i^{'} + k_{i\min} \tag{14}$$

Table I : fuzzy rules base for computing $k_p^{'}$.

Δe \ e	GN	MN	PN	ZE	PP	MP	GP
GN	G	G	G	G	G	G	G
MN	P	G	G	G	G	G	P
PN	P	P	G	G	G	P	P
ZE	P	P	p	G	p	P	P
PP	P	P	G	G	G	P	P
MP	P	G	G	G	G	G	P
GP	G	G	G	G	G	G	G

Tableau II : fuzzy rules base for computing $k_i^{'}$.

Δe \ e	GN	MN	PN	ZE	PP	MP	GP
GN	G	G	G	G	G	G	G
MN	G	G	P	P	P	G	G
PN	G	G	G	P	G	G	G
ZE	G	G	G	P	G	G	G
PP	G	G	G	P	G	G	G
MP	G	G	P	P	P	G	G
GP	G	G	G	G	G	G	G

Fig 6 shows the block diagram of the indirect field oriented control by an adaptive controller FLC-PI.

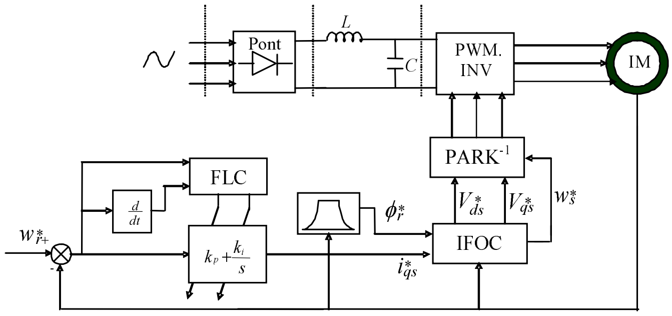


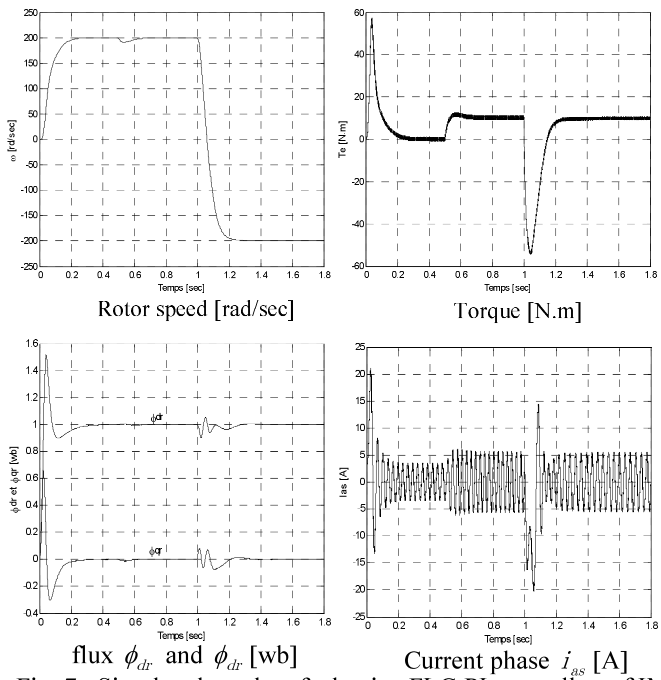
Fig. 6: Block diagram of IM control by a PI controller adapted by an FLC (adaptive FLC-PI)

C. *Results of simulation*

To prove the efficiency of the proposed method, we apply the designed controller to the control of the induction motor. The induction motor is a three phase, Y connected, four pole, 1.5 kW, 1420min^{-1} 220/380V, 50Hz. The configuration of the overall control system is shown in fig. 6. It mainly consists of an induction motor, a ramp comparison current-controlled pulse width modulated (PWM) inverter, a slip angular speed estimator, an inverse park, an outer speed feedback control loop and a fuzzy sliding mode speed controller optimized by genetic algorithm. The machine parameters are given in appendix.

Fig. 8 shows the disturbance rejection of adaptive FLC-PI controller when the machine is operated at 200 [rad/sec] under no load and a load disturbance torque (10 N.m) is suddenly applied at 1sec, followed by a consign inversion (-200 rad/sec) at 2sec. The adaptive controller rejects the load disturbance rapidly with a negligible steady state error.

Fig. 8 shows the parameters variations of PI controller during control operation.



Rotor speed [rad/sec] Torque [N.m]

flux ϕ_{dr} and ϕ_{dr} [wb] Current phase i_{as} [A]

Fig. 7 : Simulated results of adaptive FLC-PI controller of IM

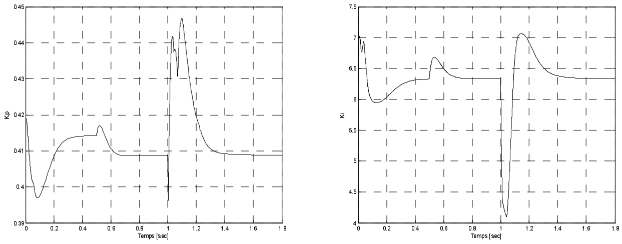


Fig. 8 : parameters variation of the adapted PI by an FLC (FLC-PI adaptive).

IV SPEED CONTROL OF IM WITH AN ADAPTIVE REGULATEUR FLC-PI OPTIMIZES BY AG

A. *Genetic Algorithms*

GA [11] are search algorithms that use operations found in natural genetic to guide through a search space. GA use a direct analogy of behavior. They work with a population of chromosomes, each one representing a possible solution to a given problem. Each chromosome has assigned a fitness score according to how good solution to the problem it is. GA are theoretically and empirically proven to provide robust search in complex spaces, giving a valid approach to problem requiring efficient and effective searching [8,13].

Any GA starts with a population of randomly generated solutions, chromosomes, and advances toward better solutions by applying genetic operators, modeled on the genetic processes occurring in nature. In these algorithms we maintain a population of solutions for a given problem; this population undergoes evolution in a form of natural selection. In each generation, relatively good solutions reproduce to give offspring that replace the relatively bad solutions which die. An evaluation or fitness function plays the role of the environment to distinguish between good and bad solutions. The process of going from the current population to the next population constitutes in the execution of GA.

Although there are many possible variants of simple GA, the fundamental underlying mechanism operates on a population of chromosomes and consists of three operations:

Evaluation of individual fitness,

Formation of gene pool (intermediate population)

Recombination and mutation.

The next procedure shows the structure of a simple GA [7,8,13].

Structure of standard genetic algorithm

```
Begin (1)
    t = 1
    Initialize Population(t)
    Evaluate fitness Population(t)
    While (Generations < Total Number) do
    Begin (2)
        Select Population(t+1) out of Population(t)
        Apply Crossover on Population(t+1)
        Apply Mutation on Population(t+1)
```

Evaluate fitness Population(t+1)
t = t + 1
End (2)
End (1)

A fitness function must be devised for each problem to be solved. Given a particular chromosome, a solution, the fitness function returns a single numerical fitness, which is supposed to be proportional to the utility or adaptation of the individual which that chromosome represents.

There are a number of ways of making this selection. We might view the population as mapping onto a roulette wheel, where each chromosome is represented by a space that proportionally corresponds to its fitness. By repeatedly spinning the roulette wheel, chromosomes are chosen using *stochastic sampling with replacement* to fill the intermediate population. The selection procedure proposed in [12], and called *stochastic universal sampling* is one of the most efficient, where the number of offspring of any structure is bound by the floor and ceiling of the expected number of offspring.

After selection has been carried out the construction of the intermediate population is complete, then the genetic operators, crossover and mutation, can occur. A crossover operator combines the features of two parent structures to form two similar offspring. It is applied with a probability of performance, the crossover probability (P_c). A mutation operator arbitrary alters one or more components of a selected structure so as to increase the structural variability of the population. Each position of each solution vector in the population undergoes a random change according to a probability defined by a mutation rate, the mutation probability (P_m).

B. Design of fuzzy-genetic system

Different approaches have been proposed to automate the design of fuzzy systems [3,7,8]. Many of these approaches take the genetic algorithm as a base of the learning process. A GA was used to optimize the fuzzy logic input membership functions, the fuzzy rules, the output membership functions and universe of discourse [3, 4].

B.1. Membership parameters optimization

GA are applied to modify the membership functions. When modifying the membership functions, these functions are parameterized with one to four coefficients (fig. 9), and each of these coefficients will constitute a gene of the chromosome for the GA.

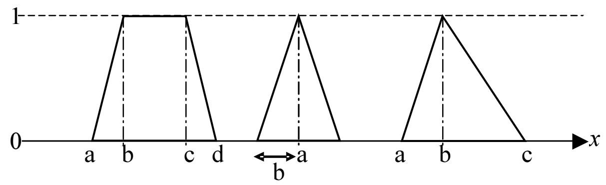


Fig. 9: Some parameterized membership functions

B.2. Fuzzy rule base optimization

Different methods are defined to apply GA to the rule base optimization, depending on its representation. For example, GA are used to modify the decision table of an FLC, which is applied to control a system with two input (trial-and-error) and one input (command action) variables. A chromosome is formed from the decision table by going row-wise and coding each output fuzzy set as an integer in 0, 1,…, *n*, where *n* is the number of membership functions defined for the output variable of the FLC. Value 0 indicates that there is no output, and value *k* indicates that the output fuzzy set has the *k*-th membership.

B.3. Algorithm of optimization by AG of the fuzzy adapter

The application of the GA in the optimization of the FL controllers can be reformulated as follows:

1. Start with an initial population of solutions that constitutes the first generation *(*P*(0))*.
2. evaluate P*(0)*:
   a) Take each chromosome (KB) from the population and introduce it into the FLC,
   b) Apply the FLC to the controlled system for an adequate evaluation period,
   c) Evaluate the behavior of the controlled system by producing a performance index to the KB.
3. While the termination condition is not met, do
   a) create a new generation *(*P*(*t+*1))* by applying the evolution operators (selection, crossover and mutation) to the individuals in P*(*t*)*,
   b) Evaluate P*(*t+*1)*
   c) t = t+*1*.
4. *End.*

The mechanism of the optimization can be represented in fig. 10.

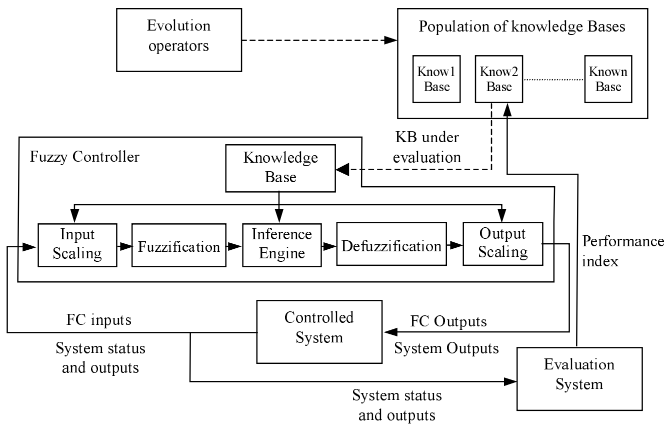


Fig. 10: Evolutionary learning of an FLC

We propose a genetic learning method for the Data Base (DB) of Mamdani fuzzy rule base system that allows us to define:

- The numbers of labels for each linguistic variable.
- The universe of discourse.
- The form of each fuzzy membership function.

The fuzzy adapter consists of two inputs (error and its derivative) and two outputs (α and β), where each input has seven membership functions. These subsets are labelled by linguistic terms such as: Zero (Z), Negative (N)... etc. We use GA to search the appropriate parameters values, and to modify the decisions table of the FLC [3,11], where the chromosome is formed from the decision table and to code each membership function by a integer number from 0 to 2, number 2 indicates the number of membership function defined for the two outputs [12]. So, we can present the equivalent code by: Small (S): 1, Big (B): 2 and No output: 0.

In GA, we only need to select some suitable parameters, such as generations, population size, crossover rate, mutation rate, and coding length of chromosome [8, 13], then the searching algorithm will search out a parameter set to satisfy the designer's specification or the system requirement. In this paper, GA will be included in the design of sliding mode fuzzy controller.

The parameters for the GA simulation are set as follows:
(1) Initial population size: 30;
(2) Maximum number of generation: 100;
(3) Crossover: Uniform crossover with probability 0.8;
(4) Mutation probability: 0.01.
In this paper, the performance is measured using the following criteria.
(5) Minimum integral of squared which is given as follows:

$$J = \int_0^t e^2 dt = \int_0^t \left(\omega_r^* - \omega_r\right)^2 dt \qquad (15)$$

Fig. 11 shows the tuning scheme of PI controller adapted by a fuzzy system where their parameters are optimized by the genetic algorithm.

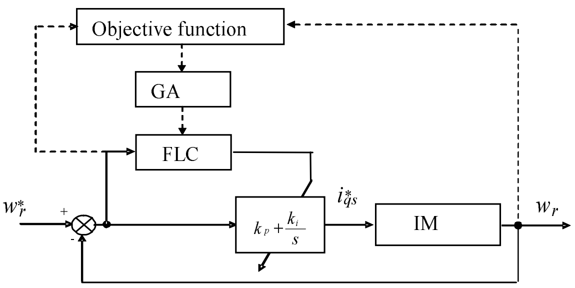


Fig 10 : the optimization technique Structure of adaptive FLC-PI by GA.

B.4. Results of optimization

The results obtained for the parameters optimization of the membership functions are represented in fig. 11 to fig. 14.

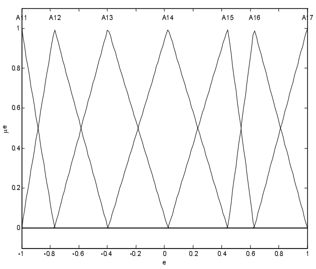


Fig. 11 : Membership function of e

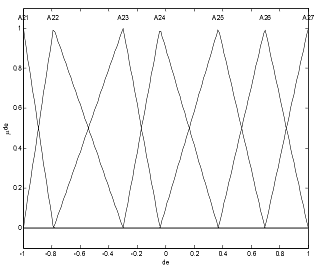


Fig. 12 : Membership functions of Δe

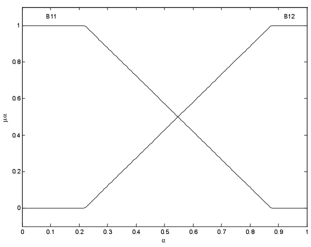


Fig. 13 : Membership function of α

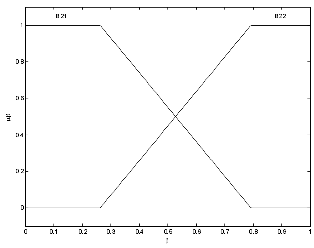


Fig. 14 : Membership function of β

The resulting rule bases from the optimization procedure are shown in table III and IV. In the tables for example, the first rule for the output α and β is:

If e is A_{11} **And** Δe is A_{21} **So** α is B_{12} and β is B_{21}

Table III: rule bases of the output α .

e \ Δe	A_{21}	A_{22}	A_{23}	A_{24}	A_{25}	A_{26}	A_{27}
A_{11}	2	1	2	2	2	2	2
A_{12}	2	1	1	2	2	2	2
A_{13}	1	2	1	2	2	1	2
A_{14}	2	2	2	2	1	2	1
A_{15}	1	1	1	2	1	2	2
A_{16}	2	2	1	2	1	2	2
A_{17}	2	2	1	2	1	2	2

Table IV: rule bases of the output β

e \ Δe	A_{21}	A_{22}	A_{23}	A_{24}	A_{25}	A_{26}	A_{27}
A_{11}	1	1	2	1	1	1	1
A_{12}	2	2	2	1	1	2	2
A_{13}	2	1	2	2	1	1	2
A_{14}	2	2	1	2	1	2	2
A_{15}	2	2	1	1	2	2	1
A_{16}	2	2	1	2	2	2	1
A_{17}	2	2	1	1	1	2	2

B.5. Results of simulation

The same tests applied for adaptive FLC-PI no optimized are applied with the adaptive FLC-PI optimized by the GA. Figure 15 illustrates the variations of PI controller parameters k_p and k_i during the optimization tests. We notice that the variations take the same trajectory for the tow controllers (adaptive FLC-PI and adaptive FLC-PI optimized). Fig. 16 shows the disturbance rejection of adaptive FLC-PI controller optimized by GA when the machine is operated at 200 [rad/sec] under no load and a load disturbance torque (10 N.m) is suddenly applied at 1sec, followed by a consign inversion (-200 rad/sec) at 2sec. This controller rejects the load disturbance very rapidly with no overshoot and with a negligible steady state error more than the adaptive FLC-PI which is shown clearly in fig. 21.

Fig. 18 shows the simulation results of the system with the adaptive FLC-PI optimized by GA in delicate conditions

such as disturbance application (T_l=5xT_{lN}) in the instant *t=0,5sec* and application of very low speed reference (w_r=*20rad/sec*) at *t=1sec*.
A test of robustness was also carried out by an increase in 300% of the rotor resistance of the machine (R_r) (fig. 19) and of 70% of its moment of inertia (J) (fig. 19). The figures show that the controller proposed gave satisfactory performances thus judges that the controller is robust.

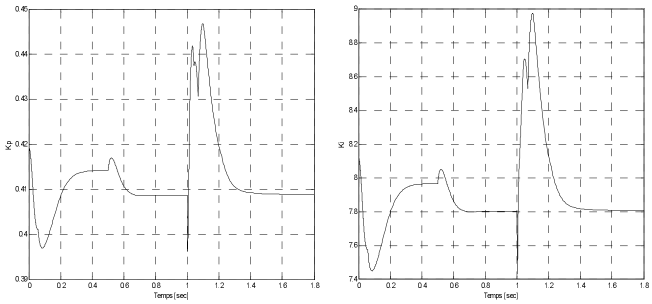
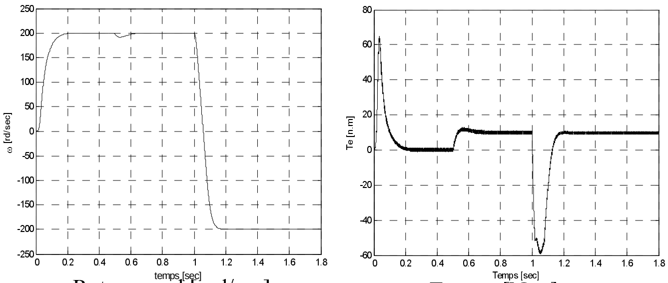


Fig. 15 : k_p and k_i values during simulation test.



Rotor speed [rad/sec] Torque [N.m]

flux ϕ_{dr} and ϕ_{dr} [wb] Current phase i_{as} [A]

Fig. 16 : Simulated results of adaptive FLC-PI controller optimized by GA of IM

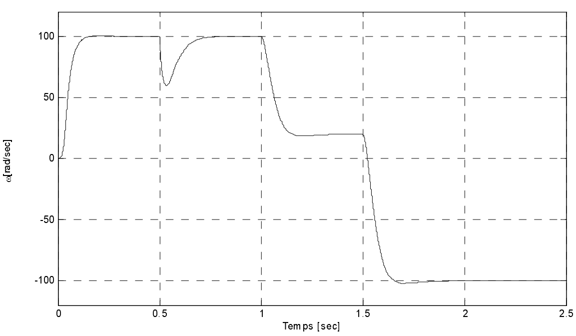


Fig. 17 : IM speed control with FLC-PI optimized by GA in delicate conditions.

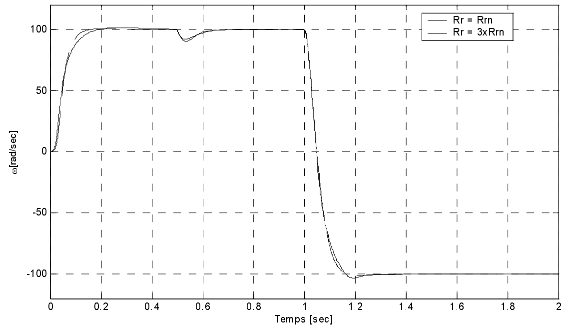


Fig. 18 : the IM rotor speed control with adaptive FLC-PI optimized by GA for tow different R_r

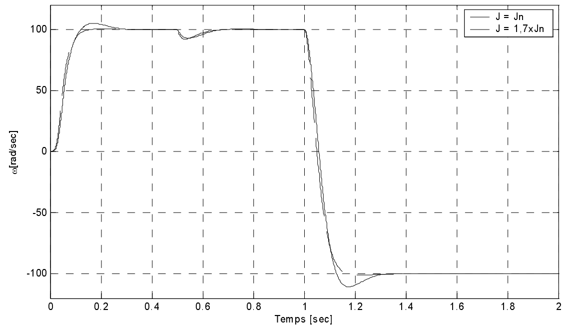


Fig. 19 : the IM rotor speed control with adaptive FLC-PI optimized by GA for tow different J.

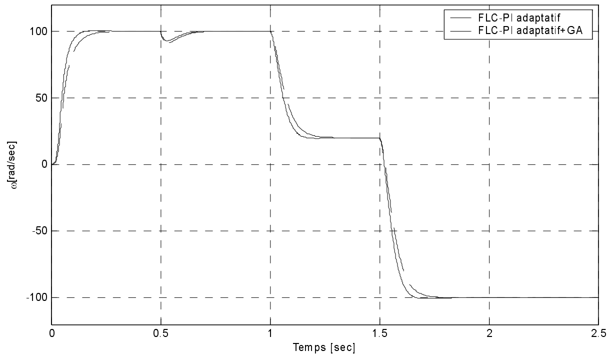


Fig. 20 : Simulated results comparison of adaptive FLC-PI and adaptive FLC-PI optimized by GA of IM.

VI CONCLUSION

In this work we proposed a method of combination between the fuzzy controller and traditional PI controller in order to overcome the disadvantages of PI controllers and FLC, this combination gave us an adaptive controller FLC-PI which presented satisfactory performances (no overshoot, minimal rise time, best disturbance rejection). The major drawback of the fuzzy controller is the insufficient analytical design technique (choice of the rules, the membership functions and the scaling factors). That we chose with the use of the genetic algorithm for the optimization of this controller in order to control IM speed. GA is used to design an adaptive FLC-PI controller with optimal parameters which present better performances compared to the adaptive FLC-PI whose parameters are chosen by the human operator.

VII APPENDIX

Induction motor parameters

P_n [Kw]	1.5	I_{an} [A]	6.31	L_s [H]	0.274
V_n [V]	220	R_s [Ω]	4.85	f_n [Hz]	50
η	0.78	R_r [Ω]	3.805	J_n [kg/m^2]	0.031
Cosφ_n	0.8	L_r [H]	0.274	f_c [N.m.s/rd]	0.008
ω_n[min^{-1}]	1428	L_m [H]	0.258	P	2

VIII. REFERENCES

[1] L. Baghli, "Contribution to Induction Machine Control: Using Fuzzy Logic, Neural Networks and Genetic Algorithms", Doctoral Thesis, Henri Poincaré University, January 1999. (Text in French.).

[2] M.A. Ouhrouche, C. Volet, ″Simulation of a Direct Field-Oriented Controller for an Induction Motor Using Matlab/Simulink Software Package″, Proc. Of IASTED Int. Conf. Modeling and Simulation (MS'2000), USA.

[3] Zhen-YU ZHAO, MOSAYOSHI TOMIZUKA, SATORU ISAKA, ″Fuzzy Gain Scheduling of PID Controllers″, IEEE Trans. on Systems, Man. and Cybernetics, Vol. 23 n°5, 1993.

[4] P. Viljamaa, H.N. Koivo, ″ Fuzzy Logic in PID Gain Scheduling″, EUFIT5, Proc. Vol. n°1, Allmagne.

[5] P. Viljamaa, ″Fuzzy Gain Scheduling and Tuning of Multivariable Fuzzy Control-Methods of Computing in Control Systems″, Technologic doctoral thesis, Temper University of tehcnology, Finland, 2000.

[6] K.C. Ng, Y.Li, D.J. Murray-Smith, K.C. Sharman, ″Genetic Algorithms Applied to Fuzzy Sliding Mode Controller Desisgn″, Proc. 1st IEE/IEEE Int. Conf. on GA in Eng. Syst., Innovation Appl. Scheffield/1995.

[7] J. R. Velesco, Louis Magedalena, ″Genetic Alogrithm in Fuzzy Control Systems″, Editor J. Periaux and G.Winter, Genetic algorithms in engineering and computer science, John Wiley and Sons, 1995.

[8] O. Cordon,F. Herrera, ″A General Study on Genetic Fuzzy Systems″, J. Periaux and G.Winter, M. Galan, P. Custa (Eds.), Genetic algorithms in engineering and computer science, John Wiley and Sons, 1995.

[9] J.P. CARON, J.P. HAUTIER, Modélisation et Commande de la Machine Asynchrone, Edition Technip, 1995.

[10] LI Zhen, Longya XU, ″On-Line Fuzzy Tuning of Indirect Field-Oriented Induction Machine Drives″, IEEE Trans. on Power Electronics, Vol. 13 n°1,1998.

[11] A. Derdiyok, M. K. Guven, Habib-Ur Rahman, N. Inane, ″Design and Implementation of New Sliding-Mode Observer for Speed-Sensorless Control of Induction Machine″. IEEE Trans. on Industriel Electronics, Vol 1. n°3, 2002.

[12] Kim Chwee NG, Yun LI, ″Design of sophisticated fuzzy logic controllers using genetic algoithms″, Proc. 3rd IEEE Int. Conf. on Fuzzy Systems, Orlando, 1994.

[13] D. E. Goldberg, "Genetic algorithms, Exploration, Optimization and Machine Learning" Addison-Wesley Edition, France, SA 1994. (Text in French).