Optimal Design of a Fuzzy Neural Network Using a Multiobjective Genetic Algorithm

Faouzi Titel and Khaled Belarbi

Abstract— **In this work, a genetic algorithm based approach for designing fuzzy inference systems (FIS) from data is developed. The FIS is implemented through an interconnected multi-layer network including neurons able to perform fundamental fuzzy operations called fuzzy neural network (FNN). Connections between neurons are weighted through binary and real weights. Then a simple and easy to implement multiobjective evolutionist algorithm is used to perform both parameters and structure optimization of the FNN by minimizing two objective functions; one objective relates to the number of rules, for compactness, while the second is the mean square error, for accuracy. In order to preserve interpretability of fuzzy rules during the optimization process, some constraints are imposed. The approach is tested on two examples: modeling of a well known temporal series and the control of the pole and cart system.**

Keywords: **Multiobjective optimization, genetic algorithms, fuzzy neural network, fuzzy inference system, Pareto optimal solutions**

I. Introduction

Fuzzy inference systems (FIS) are one of the most important applications of fuzzy logic theory; they have proved to be a very powerful technique for identifying and controlling nonlinear systems. FIS of Mamdani's type as well as Takagi-Sugeno's type has two principal characteristics. On the one hand, they are able to represent human knowledge. On the other hand, they are universal approximators [1]. Hence, these two characteristics have been used to design two kinds of FIS: expert knowledge based FIS and data based FIS. The designing approach of the first kind offers FIS with high semantic level, but they may suffer from a loss of accuracy for complex systems. Designing FIS from data can be decomposed into automatic rule generation and system optimization [2]. Fore rule generation, there exist several techniques that can be classified into three kinds. The first kind uses a grid partitioning of the multidimensional space that assigns for each variable a number of fuzzy sets, and then the fuzzy rules are generated according to these fuzzy subsets. The second kind uses fuzzy clustering in which the training data are organized into homogeneous groups and a rule is associated to each group. In the last kind, we find methods based on soft computing, the most famous and widely used being genetic algorithms and neural networks. FIS optimization can be divided into parameter and structure optimization. Parameter optimization consists of tuning the membership functions and rule conclusion while structure optimization consists in selecting variables, reducing the rule base and optimizing the number of fuzzy sets.

Genetic algorithms have become a standard approach for the design of FIS. A number of papers have been devoted to the automatic generation of the parts of a knowledge base (KB) of a fuzzy system: the data base (DB) and the rule base (RB) [3]. The decision on which part of the KB to adapt depends on two conflicting objectives: dimensionality and efficiency of the search. This obvious trade-off between the completeness and dimensionality of the search space and the efficiency of the search offers different possibilities for the design of FIS [3].

On the other hand, there is a trade-off between model accuracy and complexity in fuzzy systems [2],[4],[5]. Complexity, related to the interpretability of the fuzzy system, is determined by the compactness, considered through the number of rules in the RB, the number of input variables involved in each rule and distinguishability of the fuzzy sets.

Recent works have suggested genetic algorithms to improve both accuracy and interpretability of a known fuzzy system [4], [6], [7],[8], [9],[12],[13]. In this work, a fuzzy neural network (FNN) is adopted for implementing a Mamdani FIS. The structure of the proposed FNN has five layers. In order to guaranty completeness of fuzzy partitions, a special partitioning using triangular membership functions is adopted. A multiobjective Pareto based genetic algorithm called Niched pareto genetic algorithm (NPGA) is used for both parameter and structure optimization of the FNN. Two objectives are involved in the optimization process: the number of rules in the RB to ensure compactness and the mean square error for accuracy. This paper is organized as follows. In section II, an FNN structure is introduced. The multiobjective genetic algorithm solution procedure is presented in section III. Section IV presents the application to two examples: a modeling problem and a fuzzy controller design problem. Finally, section V gives the conclusion of this paper.

II. Structure of the Fuzzy Neural Network

This section presents a multilayer neural network, called FNN that implements a Mamdani FIS [15]. A schematic diagram of the proposed FNN structure is shown in Fig.1.

F. Titel is with Département de Génie Electrique, Faculté de Technologie, Université 20 Août 1955 Skikda, Skikda, Algérie. (corresponding author phone: +213773439522; e-mail: f_titel@yahoo.fr).

K. Belarbi is with the Department of Electronics, Faculty of Engineering, University of Constantine, Constantine, Algeria.

For the simplicity of presentation, a monovariable FIS with any number of inputs but only one output is developed.

The FNN [16] system consists of five layers: an input layer, a membership (fuzzification) layer, AND layer, OR layer and a deffuzification layer. Next we shall indicate the signal propagation and operation functions of the nodes in each layer.

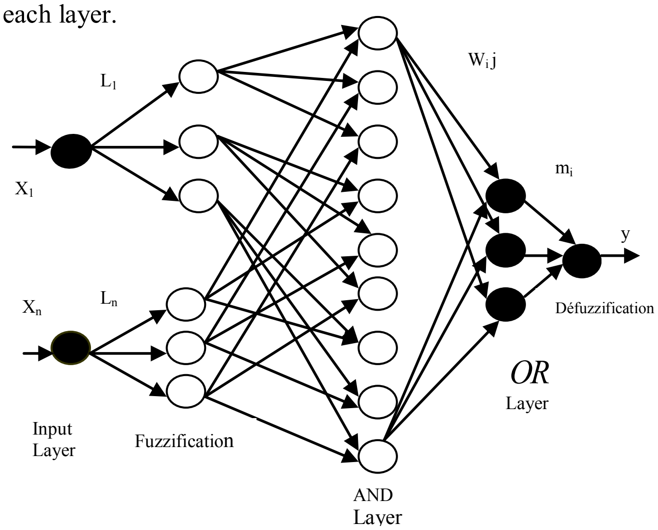

Fig. 1. Structure of the proposed FNN.

Layer1 (input layer): The nodes in this layer are input nodes with crisp input $x_i, i = 1 \dots n,$ they are transmission nodes, they only transmit input values to the next layer.

$$v_i = x_i \qquad (1)$$

Layer2 (fuzzification layer): Nodes at this layer compute the value of the membership function of inputs v_i. All nodes connected to the same input node have the same weight L_i corresponding to the central part of the universe of discourse of input variables. In order to guaranty completeness and distinguishability of fuzzy partitions, a triangular symmetric partitioning is used as shown in Fig.2. The output of node (i,j) is given by:

$$\mu_{A_{ij}}(v_i) = \begin{cases} v_i(N_i - 1/L_i) + (N_i - 1/2) - j + 2\,, \\ \qquad if\ a_{ij-1} < v_i < a_{ij} \\ -v_i(N_i - 1/L_i) - (N_i - 1/2) + j, \\ \qquad if\ a_{ij} < v_i < a_{ij+1} \\ 1\ , \qquad v_i < a_{i1}\ ou\ \ v_i > a_{iNi} \end{cases} \qquad (2)$$

With fuzzy subsets A_{ij} , $i=1, \dots, n$; $j=1, \dots, N_i$,the number of fuzzy sets associated with variable i and summits of the fuzzy sets given by:

$$a_{ij} = (-(1/2) + ((j-1)/(N_i - 1)))L_i \qquad (3)$$
$$avec\ i = 1, \dots, n\ et\ j = 1, \dots, N_i$$

Layer3 (AND layer): Each node of this layer represents a fuzzy rule. The following AND operation is applied to each rule node:

$$y_k^{And} = \mu_{A_{1j1}}(v_1).\mu_{A_{2j2}}(v_2)\dots\mu_{A_{njn}}(v_n) \qquad (4)$$

$$j_i = 1 \dots N_i \quad , i = 1 \dots n,\ k = 1 \dots \prod_{i=1}^{n} N_i$$

There are no weights to be adjusted in this layer.

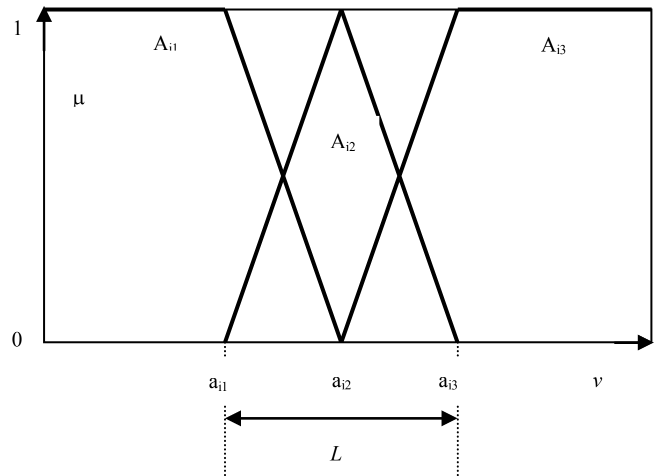

Fig. 2. Triangular symmetric portioning of universe of discourse with three fuzzy subsets.

Layer4 (OR layer): In this layer, rules with the same consequence are integrated through the fuzzy OR operation which is implemented by:

$$y_l^{OR} = 1 - \prod_{k=1}^{Na}(1 - y_k^{AND} W_{kl}),\ l = 1, \dots, N_o \qquad (5)$$

Where W_{kl} are the weights associated with node k of the AND layer and node l of the OR layer, N_a number of nodes in the AND layer and N_o the number of fuzzy sets associated with the output variable. Since one rule has only one consequence, we have:

$$W_{kl} \in \{0,1\}\ \forall\, k, l \qquad (6)$$

Layer5 (deffuzifacation layer): node at this layer realizes the deffuzification operation using the center of gravity rule

$$u = \sum_{i=1}^{No} m_i y_i^{OR} / \sum_{i=1}^{No} y_i^{OR} \qquad (7)$$

Where m_i are real weights corresponding to the centers of the triangular fuzzy sets of the output variable and can be expressed by:

$$m_i = \left(-\frac{1}{2} + \frac{i-1}{Ni-1}\right) Lo, \quad i = 1, \dots, No \qquad (8)$$

Where Lo is the width of the central part of the universe of discourse of the output variable.

III. SOLUTION USING MULTIOBJECTIVE GENETIC ALGORITHM

A. Problem definition

The previous section describes a structure of a FNN that implements a Mamdani fuzzy inference system. A multi input one output Mamdani system is composed of rules with fuzzy consequences: $A_{1j_1}, A_{2j_2}, \dots A_{nj_n}$ *and* B_k are respectively fuzzy sets associated with the fuzzy input variables and the fuzzy output variable. $w_{ik} \in \{0,1\}$ is a binary weight that models the consequence of a rule such that $w_{ik} = 1$ if rule i has consequence B_k and 0 otherwise.

Moreover if the granularity of the output fuzzy variable is M then if $w_{ik}=0$ for $k=1 \dots M$, rule i has no consequence and is not included in the rule base. The maximum possible number of rules is given by all combinations of antecedent variables fuzzy sets and is:

$$NR = N_1 \times N_2 \times ... \times N_q \times ... \times N_n, \text{ with } N_q,\ q=1...n, \quad (9)$$

the number of fuzzy sets associated with input variable x_q.

Since rule i has at most one consequence we have the constraints:

$$\sum_{k=1}^{M} W_{ik} \leq 1 \quad (10)$$

The total number of rules is thus:

$$J = \sum_{i=1}^{NR} \sum_{k=1}^{M} w_{ik} \quad (11)$$

The degrees of freedom of such a system are the number of fuzzy sets for each fuzzy variable, N_q, the binary variables w_{ik} defining the rules and the parameters Li and Lo of the triangular membership function of input and output variables respectively.

The most general modelling problem can be expressed as finding all these parameters in order to achieve a certain degree of accuracy and a compact rule base. This is formulated as two objective optimisation problems:

Find N_q, $q=1...n$, l, Li, Lo and w_{ik} so that

$$Min\left(J_1 = \frac{1}{T}\sum_{t=1}^{T}\left(y_d(t) - \hat{y}(t)\right)^2\right) \quad (12)$$

And

$$Min\left(J_2 = \sum_{i=1}^{NR}\sum_{k=1}^{l} W_{ik}\right) \quad (13)$$

$y_d(t)$ is the desired output at t,

$\hat{y}(t)$ is the output of the FNN system and T is the time horizon.

Clearly it is possible to solve this monolithic problem as a whole. However this solution procedure may lack flexibility and may not be desirable at least for two reasons. First, it leads to a quite complicated solution procedure in terms of dimensionality and data structure. Second and more importantly, it leaves no design alternatives for the decision maker.

The structure of the FIS, namely the number of fuzzy input variables and their granularity are left as alternatives to the designer who can solve the problem with different structures until a satisfactory solution is obtained, alternatively the granularity of the antecedents can be determined using some clustering techniques [3]. The problem is thus recast as finding the parameters L_i, L_o and W_{ij}.

B. Multiobjective genetic algorithm

Multiobjective genetic algorithms (MOGA) are based on the concepts of Pareto optimality which is defined in terms of dominance. Given a minimization problem with vector-valued objective function:

$$Min\ f(x) = (f_1(x), f_2(x), ..., f_m(x)) \quad (14)$$

x_1 is said to dominate x_2 iff:

$$f_i(x_1) \leq f_j(x_2), \quad \forall\ i, j \quad (15)$$

The multi-objective problem is stated as a multi-objective optimization statement, in which, the optimization implies to find a set of non-dominated solutions to approximate the pareto front, where all the solutions are pareto-optimal.

Several algorithms which exploit the parallel properties of GA for the solution of multi objective optimization have been proposed [10],[17]. The main issues are fitness assignment and selection and diversity preservation of the population [10]. This latter is usually based on the so called niching techniques while three main strategies have been proposed for fitness assignment and selection: criterion selection, aggregation selection, and ranking.

Multi-objective algorithms are used for solving the modelling problem set in the previous section. Input to the solver will be the data set for identification, or a model of the system to be controlled, the input and output variables and their granularity. We used a simple and easy to implement multiobjective genetic algorithm called NPGA algorithm [11]. This algorithm uses a specialized tournament selection based on Pareto dominance as follows:

-Two candidates (competing individuals) are randomly selected from the population.

- A comparison set of individuals (t_{dom}) forming a dominance tournament group are picked at random from population.

-Both competitors are checked for domination by the comparison set. If one of the competing individuals is dominated by any member of the set, and the other is not, then the later one is selected for reproduction. If both or neither of the two candidates is non dominated, the winner of the tournament is decided through fitness sharing: the individual with the lowest niche count (NC_i) is selected for reproduction, this parameter is calculated as follows:

$$NC_i = \sum_{j \in pop} Sh[d(i,j)] \quad (16)$$

Where $d(i,j)$ is the phenotypic distance between individuals i and j and $Sh[d]$ is a sharing function. Typically, the following triangular sharing function is used:

$$\begin{cases} Sh[d] = 1 - \dfrac{d}{\sigma_{share}}, & d \leq \sigma_{share} \\ Sh[d] = 0, & d > \sigma_{share} \end{cases} \quad (17)$$

Where σ_{share} is the niche radius set by the user.

1) *The chromosome*

The chromosome is composed of two sub chromosomes: the first contains the parameters of the membership function associated with the input and output fuzzy variables; the second contains the binary weights W_{ik} that model the rule consequence:

/Parameters of the membership function(L_1,L_2,...,L_n)/

$W_{11}W_{12}W_{13}.../W_{21}W_{22}W_{23}.../W_{31}W_{32}W_{33}.../...$

Where the sub-chain: / $w_{i1}w_{i2}w_{i3}...w_{iM}$ / $\quad w_{i,k} \in \{0,1\}$

defines the consequence of rule i and will be called a consequence sub-chain in the sequel. As mentioned above, constraint (10), only one binary weight in a given consequence sub-chain can be equal to one and if all binary weights are zero then the associated rule has no consequence and is not included in the rule base. The length of consequence sub-chain is equal to M, the granularity of the output variable and the total number of binary weights is given $N_R \times M$, N_R the number of possible rules defined above. Moreover, when using binary representation of the rules, there is no need to alter the basic definitions of the genetic operators. In this work, the membership functions are isosceles triangles uniformly distributed in the universe of discourse as shown in Fig.2. Thus, for each fuzzy variable we need only to determine the central part of the universe of discourse to deduce the uniform distribution of all fuzzy sets in the universe of discourse. This modelling will reduce considerably the length of the chromosome. Although uniform distribution is common in fuzzy controllers, it may not be the case for fuzzy modelling. Thus in this latter case, the obtained solution would not be optimal. One way to improve this solution is to relax the constraint on the membership functions and use *a posteriori* fine tuning. This fine tuning can be accomplished by a simple genetic algorithm.

2) Crossover and mutation

A two point crossover is used: the first point falls within the first sub chromosome and the second point within the second. In order to handle constraints (10) crossover and mutation in the second sub-chromosome are altered as follows:

- Crossover: the crossover point of the second sub-chromosome is enforced at the beginning of a consequence sub-chain.
- Mutation: If mutation falls in the second sub-chromosome, as usual, a one is mutated to a zero and a zero is mutated to one. However in this latter case, (10) may be violated and we may have two one's in the same consequence sub-chain, the associated rule will have two consequences. In order to keep (10) satisfied, if there are two one's in the same consequence sub-chain, the bit that was one before mutation is set to zero.

IV. SIMULATION RESULTS

Two application examples are presented: fuzzy modeling of a well known temporal series and a controller design for the pole and cart system.

A. Example 1: Modelling of a temporal series

The FIS implemented as an FNN is applied to model a well known temporal series proposed by Box and Jenkins [14]. The process is a gas furnace with a single input *u(t)* (gas flow) and a single output *y(t)* (*CO2* concentration). The data set contains 296 data points, 200 are used for training and the whole data set is used for testing. The model considered is:

$$y(t) = f\big(y(t-1), u(t-4)\big) \tag{18}$$

The structure of the FNN has two inputs *y(t-1)* and *u(t-4)* and one output *y(t)*. The granularity of the input and output variables are fixed by the user. So, both of inputs and outputs are fuzzified using three symmetric triangular membership functions: Low (L), Medium (M) and High (H), hence, there is at most, nine rules to extract if we consider all possible combinations of the input space, therefore, there is nine AND nodes and three OR nodes.

The multiobjectif optimization problem becomes:
Find the parameters L_i (L_1, L_2) and L_o; widths of the central part of the universe of discourse of the input variables *(y(t-1), u(t-4))* and output variable *y(t)* respectively. And to extract a reduced optimal set of fuzzy rules described by the weights Wij such that:

$$Min(J_1, J_2) \tag{19}$$

Where:

$$J_1 = \sum_{k=5}^{200} |y_d(k) - y(k)| \tag{20}$$

$$J_2 = \sum_{i=1}^{9}\sum_{j=1}^{3} W_{ij} \tag{21}$$

Where y_d is the desired output and y is the output of the FNN system. This problem is recast as a maximization problem:

$$Max(f_1, f_2) \tag{22}$$

Where

$$f_1 = \frac{10^5}{1+J_1} \;\; and \; f_2 = \frac{10}{1+J_2} \tag{23}$$

The parameters of the NPGA are summarized in table I. Different values of t_{dom} and σ_{share} are examined in simulation as illustrated in table I. The best results are obtained for $t_{dom} = 25 \; and \; \sigma_{share} = 1.5$ which give a small value of the mean square error defined by:

$$MSE = \frac{1}{292}\sum_{k=5}^{296}\big(y_d(k) - y(k)\big)^2 \tag{24}$$

TABLE I
NPGA PARAMETERS

Parameter	value
Generation	100
Population	150
Crossover probability	0.85
Mutation probability	0.001
Binary coding	8 bits
Chromosome length	51
Comparison set (t_{dom})	10, 15, 20, 25, 30
Niche radius (σ_{share})	0.5, 1, 1.5, 2, 2.5

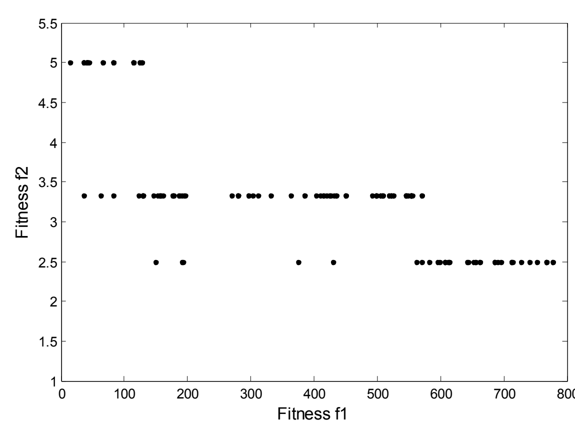

Fig. 3. Distribution of the population of the last generation.

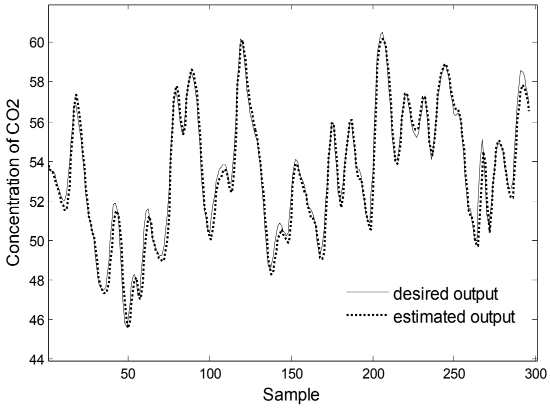

Fig. 4. The concentration of CO2.

The population of the last generation is plotted in Fig.3 which shows clearly the obtained front after 100 generations. The extracted solutions in the front are as follows:
L_1=19,6878 , L_2=35,4624 , L_o=33,2718, and
W=[001 000 000 100 010 000 000 000 000] corresponding to the three reduced fuzzy rules:
if *u(t-4) is low* **and** *y(t-1)is low* ***then*** *y(t)is high*
if *u(t-4) is Medium* **and** *y(t-1)is low* ***then*** *y(t)is Low*
if *u(t-4) is Medium* and *y(t-1)is Medium* ***then*** *y(t)is Medium*
Fig.4 shows that the quality of the model is very good.

B. Example 2: The control of the pole and cart system

The pole and cart system has been widely used as benchmark problem in the fuzzy controller design literature. The control objective is to balance the pole by applying a force on the basis of the cart. Although simple in nature, it presents some nice features for controller benchmarking: it is highly nonlinear when far from the vertical equilibrium and is sensitive to parameters variation as initial conditions, pole length and mass.

The FNN system that implements the fuzzy controller has two inputs, the pole angle $\theta(t)$ and its variation $\Delta\theta(t)$ and the output is the force F to be applied to the cart. Three symmetric triangular fuzzy membership functions (NEGATIVE(NE), ZERO(ZE) and POSITIVE(PO)) are used for both inputs and output.

The goal of the MOGA is to find optimal values of L_i (L_1,L_2) L_o and Wij such that:

$$Min(J_1, J_2) \tag{25}$$

Where:

$$J_1 = \sum_{k=1}^{500} |\theta(k)| + 10^{-2}|F(k-1)| + |\Delta\theta(k)| \tag{26}$$

$$J_2 = \sum_{i=1}^{9}\sum_{j=1}^{3} W_{ij} \tag{27}$$

This problem is recast as a maximization problem:

$$Max(f_1, f_2) \tag{28}$$

Where

$$f_1 = \frac{10^5}{1+20\,J_1} \;\; and \; f_2 = \frac{10}{1+J_2} \tag{29}$$

TABLE II
NPGA PARAMETERS

Parameter	value
Generation	100
Population	100
Crossover probability	0.9
Mutation probability	0.001
Binary coding	8 bits
Chromosome length	51
Comparison set (t_{dom})	10
Niche radius (σ_{share})	1.5

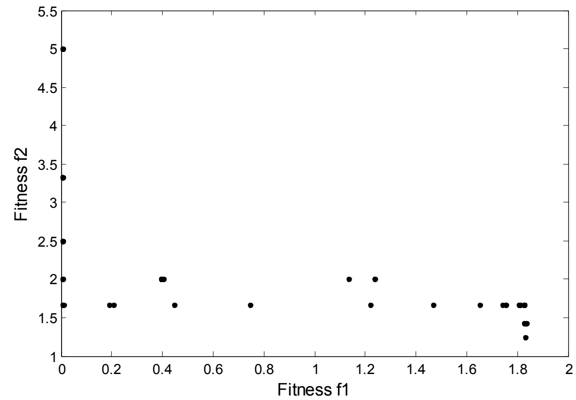

Fig. 5. Distribution of the population of the last generation.

The cost J_1 is obtained from a closed loop simulation with a nominal model having a pole with mass m=0.1 Kg and a length l=1m and a cart with mass m_c=1 Kg. The initial conditions are: $\theta(0) = 30^o \; and \; \dot{\theta}(0) = 0^o/s$.

The parameters of the NPGA are reported in table II. Fig.5 shows the whole set of solutions obtained at the last generation. The extracted solutions at the Pareto front are:
L_1=19,9686 , L_2=81,9765 , L_o=700 and
W=[000 100 100 100 010 000 010 001 000]corresponding to the six reduced fuzzy rules:
if $\theta(t)$ is NE *and* $\Delta\theta(t)$ is ZE *then* F is NE.
if $\theta(t)$ is NE *and* $\Delta\theta(t)$ is PO *then* F is NE.
if $\theta(t)$ is ZE *and* $\Delta\theta(t)$ is NE *then* F is NE.
if $\theta(t)$ is ZE *and* $\Delta\theta(t)$ is ZE *then* F is ZE.
if $\theta(t)$ is PO *and* $\Delta\theta(t)$ is NE *then* F is ZE.
if $\theta(t)$ is PO *and* $\Delta\theta(t)$ is ZE *then* F is PO.

The six rule fuzzy controller was tested for situations different from the nominal one. The results are shown in Fig.6, the controller was particularly sensitive to the pole length with success for 0.25<L<1.5m and the initial conditions with success for -5°< $\theta(0)$<55°.

V. CONCLUSION

In this work, The Mamdani fuzzy inference system is implemented through an FNN, hence; linguistic knowledge can efficiently be inserted or extracted from the network. The trade-off between compactness of the rule base and accuracy in fuzzy inference system design is cast as two objective optimization problem. A simple multiobjective genetic algorithm is used to design optimally both membership functions of the input/output variables and fuzzy rule base modeled by the binary weights of the network on which constraints are imposed in order to ensure consistency. In order to cope with these constraints, the genetic operators are perturbed. The method provides a set of solutions from which interesting solutions, belonging to the Pareto front, are extracted. The performance of this approach is verified with two examples: modeling of a well known temporal series and a controller design for the pole and cart system. For both problems, the optimization process produced compact fuzzy inference systems with high accuracy.

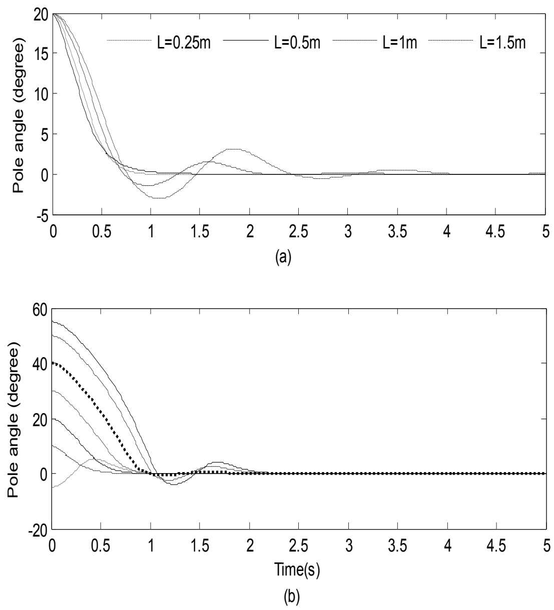

Fig. 6. The pole angle for various conditions:
(a) For different pole length.
(b) For different initial conditions.

REFERENCES

[1] L. X. Wang, "Fuzzy systems are universal approximators," Proc.1st IEE Conf. On Fuzzy systems, San Diego, 1992, pp. 1163–1169.

[2] S. Guillaume, "Designing fuzzy inference systems from data: An interpretability oriented review," *IEEE Trans. Fuzzy systems*, vol. 9, N°3, pp. 426–443, June 2001.

[3] O. Cordón, F. Gomide, F. Herrera, F. Hoffmann, and L. Magdalena, "Ten years of genetic fuzzy systems: Current framework and new trends," Fuzzy sets and systems, N°141, pp.5-31, 2004.

[4] J. Casillas, O. Cordón, M.J. del Jesus, and F. Herrera, "Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction," IEEE Trans. On Fuzzy syst., Vol. 13, N°1, pp. 13-29, Feb.2005.

[5] M. setnes, R. Babuska, and H. B. Verbruggen, "Rule-based modeling: Precision and transparency," IEEE Trans. On Systems, Man, and Cybernetics, Part C: Applications and reviews, Vol.28, N°1,pp. 165-169, Feb1998.

[6] K.Belarbi, F.Titel, W.Bourebia and K. Benmahammed, "Design of Mamdani fuzzy logic controllers with rule base minimization using genetic algorithm," Engineering applications of artificial intelligence, vol.18, N°.7, pp.875-880, October 2005.

[7] M.J. Gacto, R. Alcala and F. Herrera, "Adaptation and application of multi-objective evolutionary algorithms for rule reduction and parameter tuning of fuzzy rule-based systems," Soft Computing, Vol.13, N°5, pp.419-436, 2009.

[8] O. Cordón, F. Herrera, M.J. del Jesus, L. Magdalena, A.M. Sánchez, and P. Villar, "A multiobjective genetic algorithm for feature selection and data base learning in fuzzy-rule based classification systems," Available: http://sci2s.ugr.es/keel/pdf/keel/capitulo/villar-ipmu02.pdf

[9] M.G-Hernandez,G.I.S-Palmero,M.J.F-Aparicio,"complexity reduction and interpretability improvement for fuzzy rule systems based on simple interpretability measures and indices by bi-objective evolutionary rule selection," Soft Computing, Vol.16, N°3, pp.451-470, 2012.

[10] C.A.C. Coello, "Evolutionary multi-objective optimization: a historical view of the field," IEEE Computational intelligence magazine Vol.1, N°1, pp.28-36, Feb.2006.

[11] J. Horn, N. Nafpliotis, D.E. Goldberg, "A niched pareto genetic algorithm for multiobjective optimization," Proc.of the 1st IEEE conf. evolutionary computation, Vol.1, Orlando,FL, pp.82-87, 1994.

[12] M.Antonelli, P.Ducange, B.Lazzerini and F.Marcelloni," Multi-objective evolutionary learning of granularity, membership function parameters and rules of Mamdani fuzzy systems," Evolutionary Intelligence, Vol.2, N°1-2, pp.21-37, 2009.

[13] R.Alcala, Y.Nojima, F.Herrera, h.Ishibushi," Multiobjective genetic fuzzy rule selection of single granularity-based fuzzy classification rules and its interaction with the lateral tuning of membership functions," Soft Computing, Vol.15, N°12, pp.2303-2318, 2011.

[14] G. Box and G. Jenkins," Time series analysis: forecasting and control," Holden-Day, San Francisco, 1976.

[15] J. Buckley and Y. Hayashi," Fuzzy neural networks: a survey," Fuzzy sets and systems, Vol.66, pp.1-13, 1994.

[16] K. Belarbi and F. Titel, " Genetic algorithm for the design of a class of fuzzy controllers: An alternative approach," IEEE transactions on fuzzy systems, Vol.8, N°.4, pp.398-405, August 2000.

[17] K.Deb, A.Pratap, S.Agarwal and T.Meyarivan, "A fast and elitist multiobjective genetic algorithm:NSGA-II," IEEE Trans. On evolutionary computation, Vol.6, N°2, pp.182-197, April 2002.