Optimizing Type-1 and Type-2 Fuzzy Logic Systems with Genetic Algorithms

Nohe Ramon Cazarez-Castro[1], Luis T. Aguilar[2], Oscar Castillo[3], and Antonio Rodriguez[1]

[1] Universidad Autónoma de Baja California, Facultad de Ciencias Quimicas e Ingenieria, Tijuana BC 22390, Mexico,
nohe@ieee.org,
WWW home page: http://www.uabc.mx
[2] CITEDI-IPN, Tijuana BC 22510, Mexico
[3] Instituto Tecnologico de Tijuana, Tijuana BC 22414, Mexico

Abstract. Genetic Algorithms (GAs) are proposed as optimization method for tuning Membership's Functions (MFs) parameters of Type-1 and Type-2 Fuzzy Logic Systems (FLSs). The problem is to find the optimal MFs parameters to achieve a desired behavior in a closed-loop system. The case of study of the output regulation of a servomechanism with backlash is presented. Simulations results illustrate the effectiveness of the optimized closed-loop systems.

Key words: Fuzzy Control, Fuzzy Logic Systems, Genetic Algorithm, Type-2 Fuzzy Logic, Optimization

1 Introduction

The design of FLSs is a heavy task that we face every time that we try to use Fuzzy Logic (FL) as a solution to some problem, the design of FLSs implies at least two stages: design of rules and design of MFs.

There has been a lot work published in the design of Type-1 FLS using GA, [1] presents GAs as optimization method for control parameters, both to Type-1 Fuzzy Logic Control (FLC) as other control strategies, in [2] GAs are used to optimize all the parameters of a Type-1 FLC, in [3] a hybridizing of Neural Networks and GAs are presented to optimize a Type-1 FLC, a Hierarchical GA (HGA) is proposed in [4] to optimize rules and MFs parameters of a Type-1 FLC, in general an extended list of references can be obtained from [3] and [4].

Type-2 FLSs [5] are a generalization of Type-1 FLS [6–8], which allow us to deal with the uncertainty induced into a mechanical system by noise, frictions, backlash, etc. In the few last years a growing interest in the research of theories and applications of type-2 FLS can be seen from the academic and industry sectors. In [9] is presented an extended review of Type-2 Fuzzy Logic Systems (FLS) in control applications, and in [10] is presented a comparison between Type-1 and Type-2 FLCs, giving conditions for the use of each one.

© A. Gelbukh, M. Adiba. (Eds.)
Advances in Computer Science and Artificial Intelligence.
Research in Computing Science 39, 2008, pp. 131-153

Received 23/06/08
Accepted 01/08/08
Final version 11/09/08

As Type-2 FLSs are relatively new, and this type of FLS has more parameters to be optimized, some options have emerged to optimize some of its parameters. In [11] was proposed a Particle Swarm Optimization (PSO) method to optimize parameters of the primary MFs of Type-2 FLS. The Human Evolutionary Model (HEM) is proposed in [12] to the optimization of interval Type-2 MFs, HGAs are proposed in [13] to optimize gaussian MFs.

Type-2 FLS allow us to deal with uncertainty, but this uncertainty must to be modeled in form of Type-2 MFs, which can carry a new problem in the designing of FLCs. In [14] it is show that making a uniform modification to the MF's parameters to a certain limit, the closed-loop system keep some properties like stability, but will lost or gain in some others like performance.

GAs [15], are derivative free optimization methods that have been used in a wide range of issued, particulary in [16], GAs are presented as a class of optimization methods for FLSs.

In the present paper, the output regulation problem is studied for an electrical actuator consisting of a motor part driven by DC motor and a reducer part (load) operating under uncertainty conditions in the presence of nonlinear backlash effects, is presented as case of study. The objective is to drive the load to a desired position while providing the boundedness of the system motion and attenuating external disturbances. Due to practical requirements [17], the motor angular position is assumed to be the only information available for feedback.

The paper is organized as follows. Type-2 Fuzzy Sets and Systems are presented in Section II. A Hybrid Genetic - Fuzzy optimization approach of a Type-2 FLC is presented in Section III. Simulations results are presented in Section IV, and in Section V the conclusions are presented.

2 Fuzzy Sets and Systems

2.1 Type-1 Fuzzy Sets and Systems

A Type-1 Fuzzy Set (FS), denoted A is characterized by a type-1 MF $\mu_A(x)$ [15], where $x \in X$, i.e,

$$A = \{(x, \mu(x))|\forall x \in X\} \tag{1}$$

where $\mu(x)$ is called *membership function* of the fuzzy set A. The MF maps each element of X to a membership grade (or membership value) between 0 and 1.

Type-1 FLSs are both intuitive and numerical systems that maps crisp inputs into a crisp output. Every Type-1 FLS is associated with a set of rules with meaningful linguistic interpretations, such as:

$$R^l : \text{IF } x_1 \text{ is } A_1^l \text{ AND } x_2 \text{ is } A_2^l \text{ THEN } w \text{ is } B^l, \tag{2}$$

which can be obtained either from numerical data, or from experts familiar with the problem at hand. Based on this kind of statements, actions are combined with rules in an antecedent/concecuent format, and then aggregated according to approximate reasoning theory, to produce a nonlinear mapping from input space $U = U_1 \times U_2 \times \ldots U_n$ to output space W, where $A_k^l \subset U_k$, $k = 1, 2, \ldots n$, and the output linguistic variable is denoted by w.

A Type-1 FLS consist of four basic elements (see Fig. 1): the ***Type-1 fuzzyfier***, the ***fuzzy rule-base***, the ***inference engine***, and the ***Type-1 defuzzifier***. The ***fuzzy rule-base*** is a collection of rules in the form of (2), which are combined in the ***inference engine***, to produce a fuzzy output. The ***Type-1 fuzzifier*** maps the crisp input into Type-1 (FS), which are subsequently used as inputs to the ***inference engine***, whereas the ***Type-1 defuzzifier*** maps the Type-1 FSs produced by the ***inference engine*** into crisp numbers.

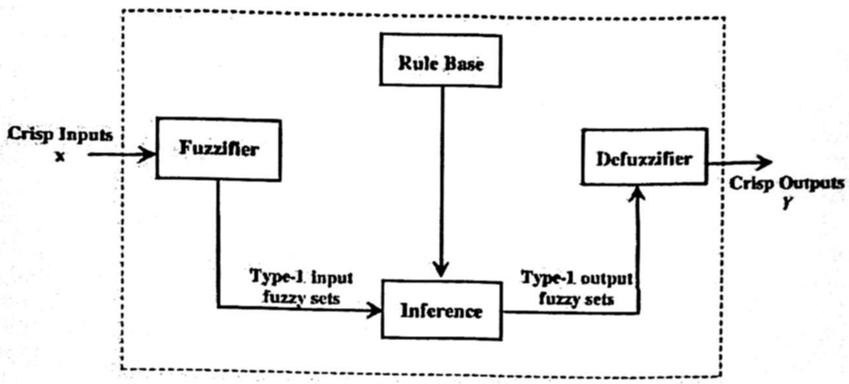

Fig. 1. Type-1 Fuzzy Logic System.

FSs can be interpreted as MFs U_x that associate with each element of x of the universe of discourse, U, a number $\mu_X(x)$ in the interval [0,1]:

$$\mu_x : U \to [0, 1]. \tag{3}$$

2.2 Type-2 Fuzzy Sets and Systems

As the Type-1 FS, the concept of Type-2 FS was introduced by Zadeh [6–8] as an extension of the concept of an ordinary FS (Type-1 FS).

A Type-2 FS, denoted $\tilde{A}$ is characterized by a Type-2 MF $\mu_{\tilde{A}}(x, u)$ [18], where $x \in X$ and $u \in J_x \subseteq [0, 1]$, i.e,

$$\tilde{A} = \{((x, u), \mu_{\tilde{A}}(x, u)) | \forall x \in X, \forall u \in J_x \subseteq [0, 1]\} \tag{4}$$

in which $0 \leq \mu_{\tilde{A}}(x, u) \leq 1$. $\tilde{A}$ can alse be expressed as follows [18]:

$$\tilde{A} = \int_{x \in X} \int_{u \in J} \mu_{\tilde{A}}(x,u)/(x,u) \tag{5}$$

where $J_x \subseteq [0,1]$ and $\int\int$ denotes union over all admissible x and u [18].

J_x is called primary membership of x, where $J_x \subseteq [0,1]$ for $\forall x \in X$ [18]. The uncertainty in the primary memberships of a Type-2 FS $\tilde{A}$, consists of a bounded region that is called the *footprint of uncertainty* (FOU) [18]. It is the union of all primary memberships [18].

A FLS described using at least one Type-2 FS is called a Type-2 FLS. Type-1 FLS are unable to directly handle rule uncertainties, because they use Type-1 FSs that are certain. On the other hand, Type-2 FLSs, are very useful in circumstances where it is difficult to determine an exact, and measurement uncertainties [5].

It is known that Type-2 FS let us to model and to minimize the effects of uncertainties in rule-based FLS. Unfortunately, Type-2 FSs are more difficult to use and understand that Type-1 FSs; hence, their use is not widespread yet.

Similar to a Type-1 FLS, a Type-2 FLS includes ***Type-2 fuzzifier***, ***rule-base***, ***inference engine*** and substitutes the ***defuzzifier*** by the ***output processor***. The ***output processor*** includes a ***type-reducer*** [5] and a ***Type-2 defuzzifier***; it generates a Type-1 FS output (from the *type-reducer*) or a crisp number (from the ***Type-2 defuzzifier***). A Type-2 FLS is again characterized by IF-THEN rules, but its antecedent and consequents sets are now of the Type-2, see (6). Type-2 FLSs can be used when the circumstances are too uncertain to determine exact membership grades. A model of a Type-2 FLS is shown in Fig. 2.

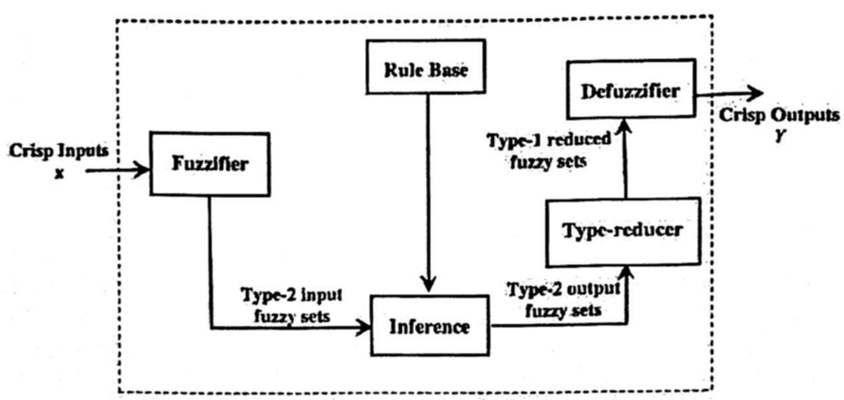

Fig. 2. Type-2 Fuzzy Logic System.

$$R^l : \text{IF } x_1 \text{ is } \tilde{A}_1^l \text{ AND } x_2 \text{ is } \tilde{A}_2^l \text{ THEN } w \text{ is } \tilde{B}^l, \tag{6}$$

Note that for both Type-1 FLS and Type-2 FLS ***rule-base*** we are describing the IF-THEN rules following the Mamdani's [19] type.

3 Genetic Algorithms

GAs are derivative-free optimizations methods based on the concepts of natural selection and evolutionary process [15]. They were first proposed and investigated in [20]. As a general-purpose optimization tool, GAs are moving out of academic sectors and finding significant applications in many areas. Their popularity can be attributed to their freedom from dependence on functional derivatives and their incorporation of other characteristics reported in [15].

The main idea of a GA is to maintain a ***population*** of solutions of a problem that evolves over a time through a process of competition and controlled variation. Each individual in the population represents a candidate solution to the specific problem, and each individual has associated a *fitness* to determine which individuals are used to form (by sexual reproduction and mutation) new ones in the process of competition.

The sexual reproduction of GAs consists basically in a **Selection Process** [20], where a set of individuals are selected to be passed through a **Crossover Operation** [20], which consist in to take a pair of individuals and interchanging its gens from one (or more) random selected cross point to the end of the chromosome. **Mutation** [20] consists in to change one or more randomly selected gens of the chromosome in some of the selected individuals.

The ***objective function*** [20] of a GA is a the value (*fitness*) that the method must maximize or minimize.

4 Case of Study: Backlash Problem

In the present paper, the output regulation problem is studied for an electrical actuator consisting of a motor part driven by DC motor and a reducer part (load) operating under uncertainty conditions in the presence of nonlinear backlash effects. The objective is to drive the load to a desired position while providing the boundedness of the system motion and attenuating external disturbances. Due to practical requirements [17], the motor angular position is assumed to be the only information available for feedback.

The study is motivated by a PEGASUS robot manipulator (see Fig. 3) installed in the Robotics & Control Laboratory of CITEDI-IPN where the backlash problem occurs due to the chains and gears transmission elements. Measurements are provided from the motor side while the links, attached into the load of the motor, must be positioned at the desired point.

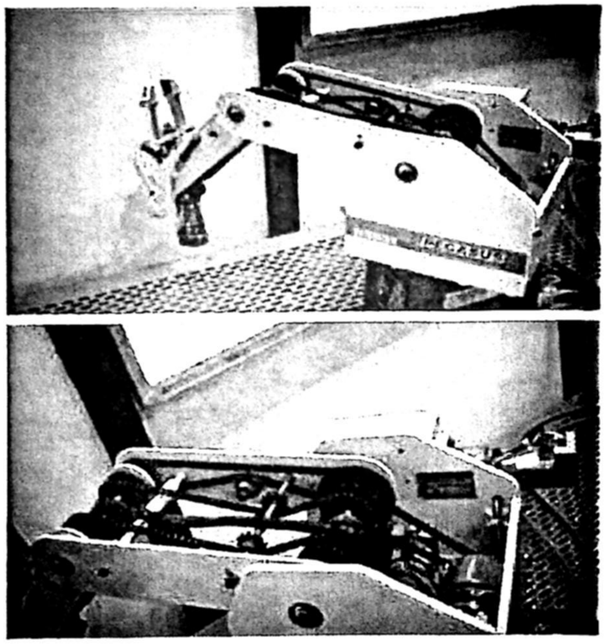

Fig. 3. PEGASUS robot manipulator of Robotics & Control Laboratory of CITEDI-IPN, and view of the problem in question given by each degree of freedom.

4.1 Dynamic Model

The dynamic model of the angular position $q_i(t)$ of the DC motor and $q_o(t)$ the angular position of the load are given according to

$$\begin{aligned} J_0 N^{-1}\ddot{q}_0 + f_0 N^{-1}\dot{q}_0 &= T + w_0 \\ J_i\ddot{q}_i + f_i\dot{q}_i + T &= \tau_m + w_i, \end{aligned} \tag{7}$$

hereafter, J_0, f_0, $\ddot{q}_0$ and $\dot{q}_0$ are, respectively, the inertia of the load and the reducer, the viscous output friction, the output acceleration, and the output velocity. The inertia of the motor, the viscous motor friction, the motor acceleration, and the motor velocity denoted by J_i, f_i, $\ddot{q}_i$ and $\dot{q}_i$, respectively. The input torque τ_m serves as a control action, and T stands for the transmitted torque. The external disturbances $w_i(t)$, $w_0(t)$ have been introduced into the driver equation (7) to account for destabilizing model discrepancies due to hard-to-model nonlinear phenomena, such as friction and backlash.

The transmitted torque T through a backlash with an amplitude j is typically modeled by a dead-zone characteristic [21]:

$$T(\Delta q) = \begin{cases} 0 & |\Delta q| \leq j \\ K\Delta q - Kj\mathrm{sgn}(\Delta q) & \text{otherwise} \end{cases} \tag{8}$$

with

$$\Delta q = q_i - Nq_0, \tag{9}$$

where K is the stiffness, and N is the reducer ratio. Such a model is depicted in Fig. 4. Provided the servomotor position $q_i(t)$ is the only available measurement on the system, the above model (7)-(9) appears to be non-minimum-phase because along with the origin the unforced system possesses a multivalued set of equilibria (q_i, q_0) with $q_i = 0$ and $q_0 \in [-j, j]$.

To avoid dealing with a non-minimum-phase system, we replace the backlash model (8) with its monotonic approximation:

$$T = K\Delta q - K\eta(\Delta q) \tag{10}$$

where

$$\eta = -2j\frac{1 - \exp\left\{-\frac{\Delta q}{j}\right\}}{1 + \exp\left\{-\frac{\Delta q}{j}\right\}}. \tag{11}$$

Coupled to the drive system (7) subject to motor position measurements, it is subsequently shown to continue a minimum phase approximation of the underlying servomotor, operating under uncertainties $w_i(t)$, $w_0(t)$ to be attenuated. As a matter of fact, these uncertainties involve discrepancies between the physical backlash model (8) and its approximation (10) and (11).

4.2 Problem statement

To formally state the problem, let us introduce the state deviation vector $x = [x_1, x_2, x_3, x_4,]^T$ with

$$\begin{aligned} x_1 &= q_0 - q_d \\ x_2 &= \dot{q}_0 \\ x_3 &= q_i - Nq_d \\ x_4 &= \dot{q}_i \end{aligned}$$

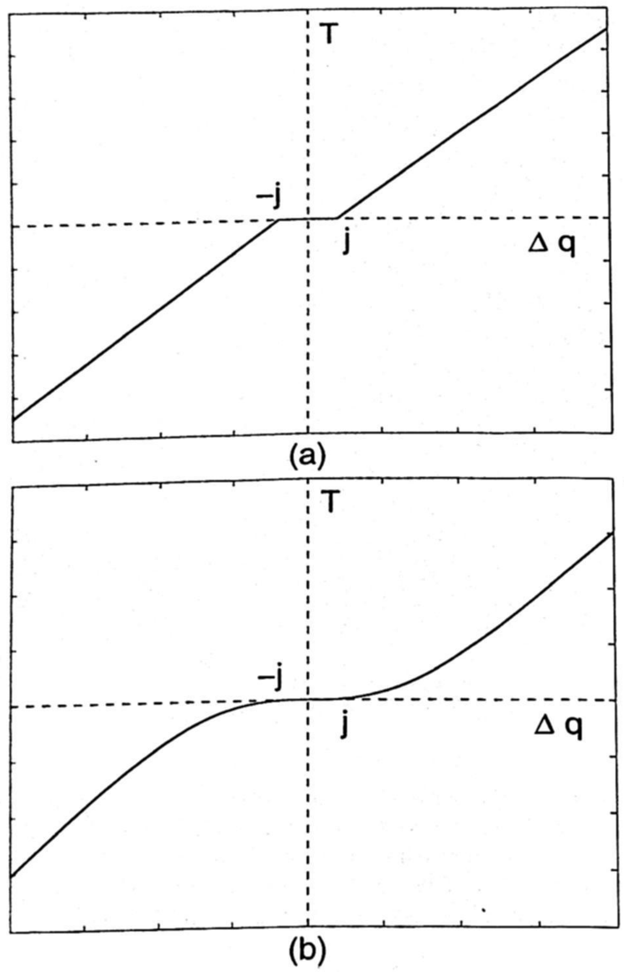

Fig. 4. a) The dead-zone model of backlash; b) The monotonic approximation of the dead-zone model.

where x_1 is the load position error, x_2 is the load velocity, x_3 is the motor position deviation from its nominal value, and x_4 is the motor velocity. The nominal motor position Nq_d has been pre-specified in such a way to guarantee that $\Delta q = \Delta x$, where

$$\Delta x = x_3 - Nx_1. \tag{12}$$

Then, system (7)-(11), represented in terms of the deviation vector x, takes the form

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= J_0^{-1}[KNx_3 - KN^2x_1 - f_0x_2 + KN\eta(\Delta q) + w_o] \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= J_i^{-1}[\tau_m + KNx_1 - Kx_3 - f_ix_4 + K\eta(\Delta q) + w_i].
\end{aligned} \tag{13}$$

The zero dynamics

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= J_0^{-1}[-KN^2x_1 - f_0x_2 + KN\eta(-Nx_1)]
\end{aligned} \tag{14}$$

of the undisturbed version of system (6) with respect to the output

$$y = x_3 \tag{15}$$

is formally obtained by specifying the control law that maintains the output identically zero.

The objective of the Type-2 FLC output regulation of the nonlinear driver system (7) with backlash (10) and (11), is thus to design a FLC so as to obtain the closed-loop system in which all these trajectories are bounded and the output $q_0(t)$ asymptotically decays to a desired position q_d as $t \to \infty$ while also attenuating the influence of the external disturbances $w_i(t)$ and $w_0(t)$.

5 Fuzzy - Genetic Architectures

In this paper we use a GA to optimize the parameters of the MFs of a Type-1 FLS and a Type-2 FLS, this optimization is performed assuming that each FLS have a preestablished fuzzy ***rule-base***.

By the knowledge that we have about the systems of our case of study, we propose the seven rules of Table 1, where can be seen that we select two input variables (`error` and `change of error`) and one output variable (`control`), each one of this input and output variables are granulated in three linguistic interpretations (MFs), this linguistic interpretations are: *negative* (n), *zero* (z) and *positive* (p).

The next two subsections describe the architectures for each one of the optimization approaches.

5.1 Fuzzy - Genetic Architecture for Type-1 FLS

To make the Type-1 Fuzzy - Genetic optimization we are considering triangular MFs to each one of the linguistic interpretation in which we granulate the three variables of the FLS, we encode each individual in a 27 real gens chromosome [22], where are represented the three parameters of each one of the three MFs

Table 1. Fuzzy IF-THEN rules

No.	error	change of error	control
1	n	n	p
2	n	p	z
3	n	z	p
4	p	p	n
5	p	n	z
6	p	z	n
7	z	z	z

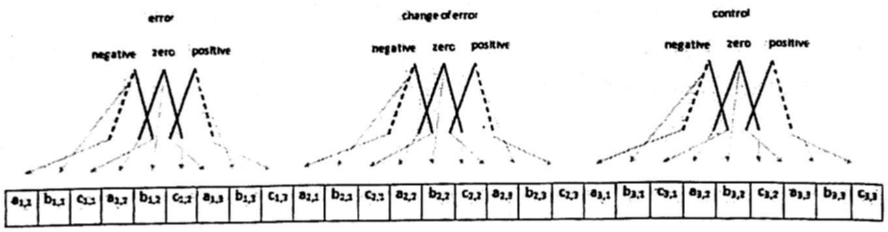

Fig. 5. Genotype for Type-1 FLS.

of each one of the three variables of the Type-1 FLS, this is called a genotype [22] of the population, see Fig. 5 for an schematic representation.

In this case we select the ***objective function*** of equation (16). To achieve our optimization problem, we must to minimize this ***objective function***.

$$\text{fitness}_i = min(mean|error|) \tag{16}$$

The set of parameters of the GA are shown in Table 2.

5.2 Fuzzy - Genetic Architecture for Type-2 FLS

To make the Type-2 Fuzzy - Genetic optimization we are considering Triangular Interval MFs to each one of the linguistic interpretation in which we granulate the three variables of the FLS; following the proposed in [23]-[24] we need six parameters for each Triangular Interval Type-2 MF, that is, we need to encode a total of 54 parameters for each individual (Type-2 FLS) of our population, to make this encoding we design a chromosome structure of 54 consecutive real gens, where are represented the six parameters of each one of the three Triangular Interval Type-2 MFs of each one of the three variables of each Type-2 FLS (individual) of our population. Fig. 6 show an schematic of the genotype [15] of our Type-2 Fuzzy - Genetic optimization approach, where the subindex of each gen represents the parameter number of the MF in question and the superindex represent the number of the gen in question, remember that each Triangular

Table 2. Parameters of the Genetic Algorithm

Parameter	Value
Representation	real
Population size	10
Selection method	Roulette [15]
Cross method	two points
Rate of cross	0.8
Mutation method	Gaussian
Rate of mutation	0.1
Elitism [15]	2
Generations	100

Interval Type-2 MF is represented by six parameters (gens) which means the left hand side of Fig. 6 represents the MF *negative* of input variable **`error`**, and the right hand side of Fig. 6 represents the MF *positive* of the output variable `control`.

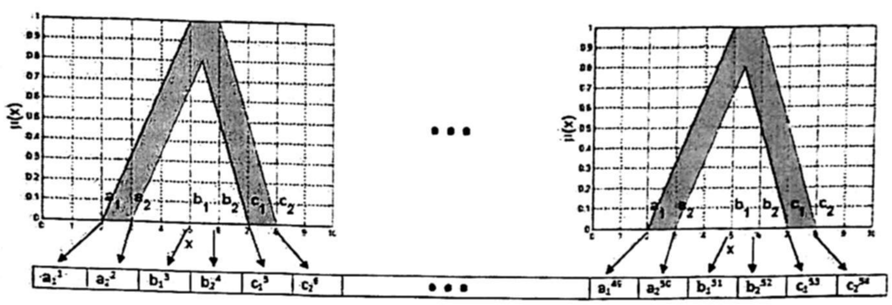

Fig. 6. Genotype for Type-2 FLS.

Our optimization problem is again a minimization, and with the ***objective function*** [15] of equation (16) we express that we want to minimize the mean error in our solutions space.

The set of the GA parameters are the same shown in Table 2.

6 Simulations Results

To perform simulations we use the dynamical model (7) of the experimental testbed installed in the Robotics & Control Laboratory of CITEDI-IPN (see Fig. 7), which involves a DC motor linked to a mechanical load through an imperfect contact gear train [21]. The parameters of the dynamical model (7) are in Table

3, while $N = 3$, $j = 0.2$ [rad], and $K = 5$ [N-m/rad]. These paremeters are taken from the experimental testbed.

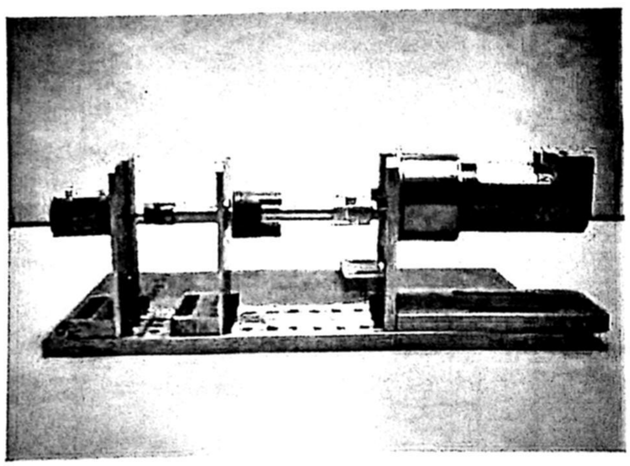

Fig. 7. Experimental test bench.

Table 3. Nominal parameters.

Description	Notation	Value	Units
Motor inertia	J_i	2.8×10^{-6}	Kg-m^2
Load inertia	J_o	1.07	Kg-m^2
Motor viscous friction	f_i	7.6×10^{-7}	N-m-s/rad
Load viscous friction	f_o	1.73	N-m-s/rad

The input-output motion graph of Fig. 8 reveals the gear backlash effect of the system.

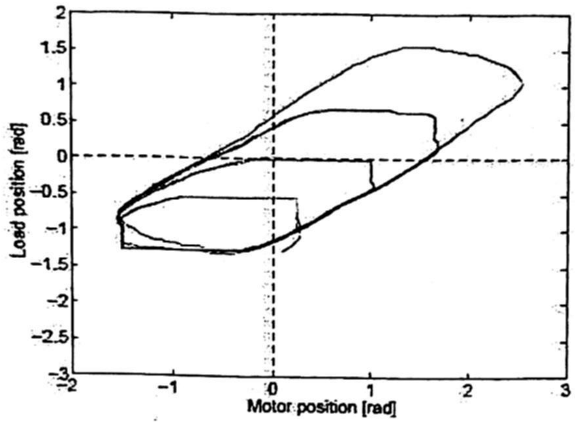

Fig. 8. Backlash hysteresis before compensation.

6.1 Numerical Solutions for the Type-1 FLS - Genetic Architecture

The GA was implemented using the Genetic Algorithm and Direct Search Toolbox [25], each individual (Type-1 FLS) of the population was tested in a closed-loop system modeled in Simulink, in that model we consider the angular motor position as the only information available for feedback. In the simulations, the load was required to move from the initial static position $q_0(0) = 0$ [rad] to the desired position $q_d = \pi/2$ [rad]. In order to illustrate the size of the attraction domain, the initial load position was chosen reasonably far from the desired position. The GA was executed in a PC Computer with Intel Pentium processor of 2.4 GHz and 512 Mb of RAM. Four executions conclude satisfactory and the solutions are concentrated in Table 4.

The GA was executed in about 26 hours (see Fig. 9 for details of convergence), producing the results shown in Table 4, Table 4 include the settling time of each individual, and as can be seen, all the individuals converge to a same fitness, this can be caused because there are just 27 gens in the chromosome (see Fig. 5) and the **Crossover Operation** makes than all individuals merge in the same one, moreover, the ***objective function*** (16) is not designed to considerate this case of situations.

Table 5 shows the chromosome of the resulting individual of Table 4, and its phenotype is in Fig. 10, its surface of control is in Fig. 11 and Fig 12 shows the system's response for this individual.

![h]

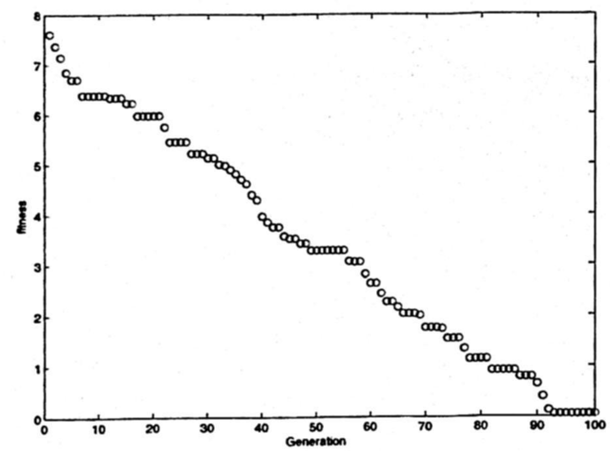

Fig. 9. Evolution of GA for the Type-1 FLS - Genetic Architecture.

Table 4. Genetic Algorithm Results for the *best* execution of the Type-1 FLS.

Individual	Fitness	Settling Time
1	0.0519	38.2158
2	0.0519	38.2158
3	0.0519	38.2158
4	0.0519	38.2158
5	0.0519	38.2158
6	0.0519	38.2158
7	0.0519	38.2158
8	0.0519	38.2158
9	0.0519	38.2158
10	0.0519	38.2158

Table 5. Data of the *best* individual of the Type-1 FLS - Genetic Architecture.

Variable	Membership Function	a	b	c
error	negative	-1.5000	-1.0000	-0.5377
	zero	-0.1173	-0.1173	2.3416
	positive	-0.5380	1.0000	1.5000
change of error	negative	-1.5000	-1.0000	0.0258
	zero	-0.4957	0.6905	0.6905
	positive	-0.5658	1.0000	1.5000
control	negative	-1.5000	-1.0000	-0.6166
	zero	-0.6619	0.7550	0.7550
	positive	0.0800	1.0000	1.5000

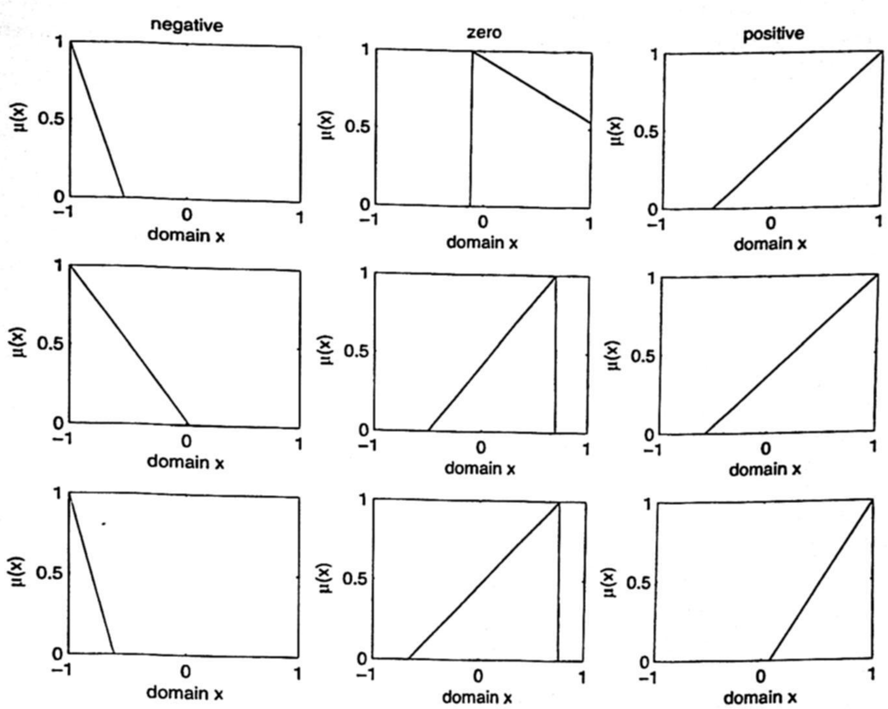

Fig. 10. Phenotype of the *best* individual of the Type-1 FLS - Genetic Architecture.

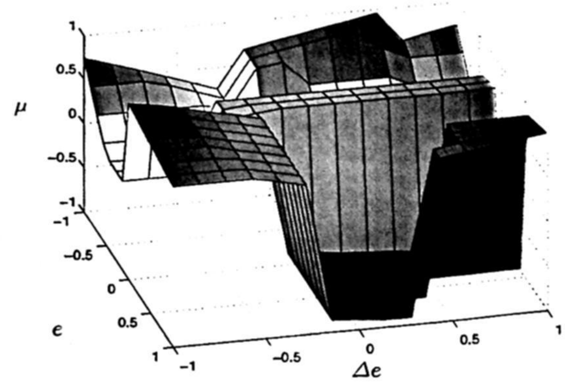

Fig. 11. Surface of control of the *best* individual of the Type-1 FLS - Genetic Architecture.

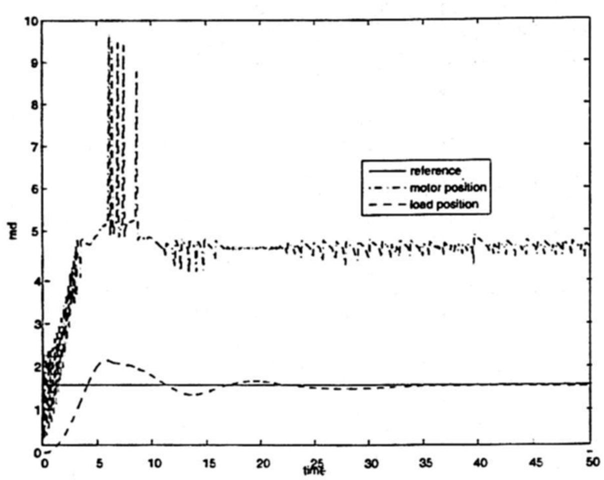

Fig. 12. Simulation result of the *best* individual of the Type-1 FLS - Genetic Architecture.

6.2 Numerical Solutions for the Type-2 FLS - Genetic Architecture

The GA was implemented using the Genetic Algorithm and Direct Search Toolbox [25], each individual (Type-2 FLS) was implemented in the Interval Type-2 Fuzzy Logic Toolbox reported in [23]-[24], and each individual of the population was tested in a closed-loop system modeled in Simulink, in that model we consider the angular motor position as the only information available for feedback. In the simulations, the load was required to move from the initial static position $q_0(0) = 0$ [rad] to the desired position $q_d = \pi/2$ [rad]. In order to illustrate the size of the attraction domain, the initial load position was chosen reasonably far from the desired position. The GA was executed in a PC Computer with Intel Pentium processor of 2.4 GHz and 512 Mb of RAM.

The GA was executed in about 200 hours (see Fig. 13 for details of convergence), producing the results shown in Table 6, including the settling time of each individual, and as can be seen, a best fitness do not mean a best settling time, this because the ***objective function*** (16) is not designed to achieve this performance measurement, moreover, some of this results can give a good performance in simulation, but in physical applications can be destructive for the mechanisms.

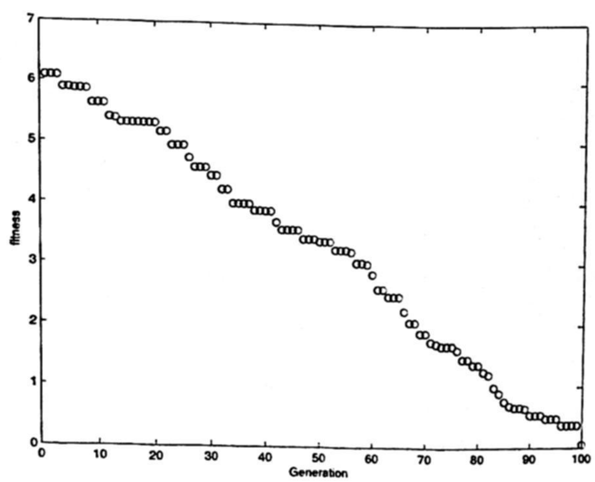

Fig. 13. Evolution of GA for the Type-2 FLS - Genetic Architecture.

From Table 6, the individual number six is the best solution of the GA, this because it have the best fitness, Table 7 shows the chromosome of individual number six who's phenotype [15] that is depicted in Fig. 14. Fig. 15 shows its surface of control and Fig. 16 shows the system's response for this individual number six.

Table 6. Genetic Algorithm Results of the Type-2 FLS.

Individual	Fitness	Settling Time
1	0.3032	17.1036
2	0.8898	24.1193
3	0.8593	15.3227
4	0.3046	17.0761
5	0.3809	16.0508
6	0.3004	15.2611
7	0.3308	16.4957
8	2.2201	22.9921
9	0.6113	20.8435
10	2.3717	35.2812

Table 7. Data of the *best* individual of the Type-2 FLS - Genetic Architecture.

Variable	μ	a1	b1	c1	a2	b2	c2
	negative	-1.0000	0.2176	0.2176	0.3012	0.4293	0.4293
error	zero	-0.1982	-0.1982	-0.1912	-0.4260	0.8378	0.8378
	positive	0.2669	0.2669	1.0000	-0.5479	-0.5479	1.0000
	negative	-1.0000	0.3504	0.6897	0.0926	0.9659	0.9659
change of error	zero	-0.6963	-0.6963	-0.1490	-0.4170	0.0000	0.4005
	positive	0.2611	0.4405	1.0000	-0.9561	-0.9561	1.0000
	negative	-1.0000	0.4586	0.4586	-0.6673	0.7247	0.7247
control	zero	-0.4571	-0.4571	0.7067	0.0446	0.0446	0.0446
	positive	0.6220	0.9432	1.0000	-0.0605	-0.0605	1.0000

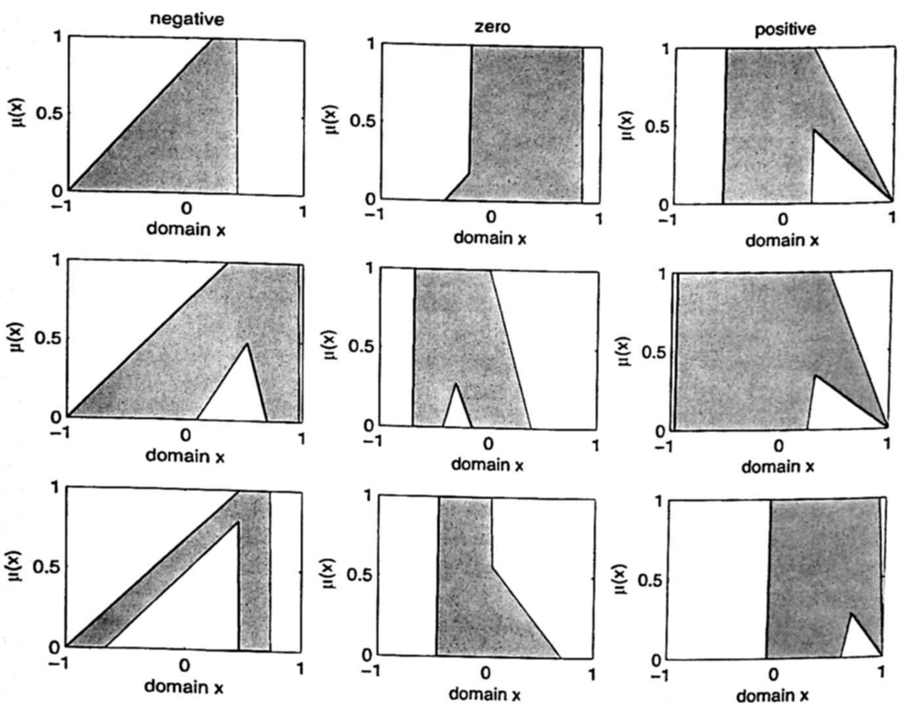

Fig. 14. Phenotype of the *best* individual of the Type-2 FLS - Genetic Architecture.

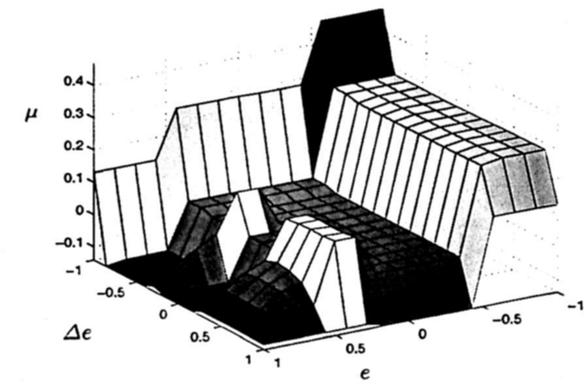

Fig. 15. Surface of control of the *best* individual of the Type-2 FLS - Genetic Architecture.

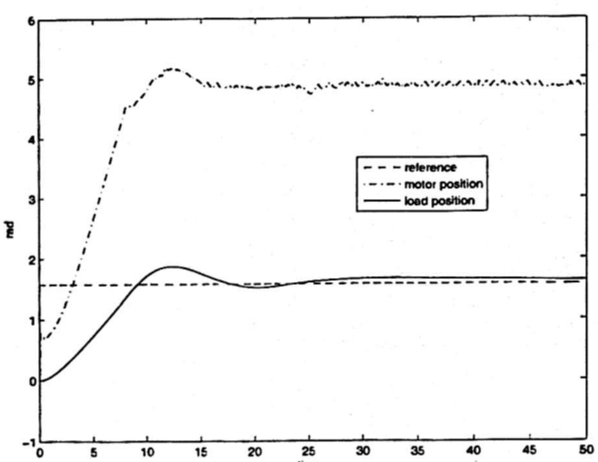

Fig. 16. Simulation result of the *best* individual of the Type-2 FLS - Genetic Architecture.

7 Conclusion

The main goal of this paper was to develop hybrid approaches combining GAs with Type-1 and Type-2 FLSs for the design and optimization of Type-1 and Type-2 FLCs. The optimized Type-1 and Type-2 FLCs were designed for the case of study of the output regulation of a servomechanism with backlash, giving to the GA a heavy task because the nonlinearity of the proposed problem, and a heavy task test to the Type-1 and Type-2 FLCs for the high uncertainty in the case of study.

The GA was implemented and performed in a satisfactory fashion, giving a whole family of solutions of Type-1 and Type-2 FLCs to the problem in question, some of the resulting solutions give us best performance that other ones.

In this paper we are reporting explicitly just one of the solutions for Type-1 and Type-2 FLCs because of space limitations. The solutions reported in this paper are evidently different each from another, confirming that as in the Type-1 FLS, Type-2 FLS can be obtained from human expertise, and in this case, each solution can be a representation of the expertise of different experts.

The combination of a GA and Type-1 and Type-2 FLSs results to be a good method for the proposed problem, but the time necessary to run each generation of the GA is too big, and maybe we can find another set of solutions faster by trial and error, but those results most probably will not be optimal.

If we compare the resulting best optimized Type-1 and Type-2 FLCs, is evidently that the Type-2 FLC is better that the Type-1 FLC, this because from 12 and 16 we can see that the response of the closed loop system is more soft from optimized Type-2 that from optimized Type-1 FLCs, and in 12 we can see that the optimized Type-1 FLC is giving a motor behavior that may be destructive to the physical mechanism and this do not occur with the optimized Type-2 FLC. Moreover, the settling time of the response of the closed-loop systems with the optimized Type-1 FLC is twice as long that the responte of of the closed-loop systems with the optimized Type-2 FLC.

Comparing the necessary time to the convergence of the Type-1 and Type-2 FLCs approaches, optimize the Type-2 FLC needs almost eight times that the time needs to optimize the Type-1 FLC.

GAs has been proved in this approach that are a good method to optimize MF's parameters in the designing of Type-1 and Type-2 FLSs, however, we must be aware of the price we pay for using GA, that is, we must to be agree in spend so much computation resources, like memory and CPU time.

References

1. Grefenstette, J.: Optimization of control parameters for genetic algorithms. IEEE Trans. Syst. Man Cybern. **16** (1986) 122–128
2. Lee, M., Takagi, H.: Integrating design stage of fuzzy systems using genetic algorithms. Fuzzy Systems, 1993., Second IEEE International Conference on (1993) 612–617 vol.1

3. Melin, P., Castillo, O.: Intelligent control of complex electrochemical systems with a neuro-fuzzy-genetic approach. IEEE Transactions on Industrial Electronics **48** (Oct 2001) 951–955
4. Castillo, O., Lozano, A., Melin, P.: Hierarchical genetic algorithms for fuzzy system optimization in intelligent control. Fuzzy Information, 2004. Processing NAFIPS '04. IEEE Annual Meeting of the **1** (27-30 June 2004) 292–297 Vol.1
5. Mendel, J.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice Hall, Upper Saddle River, NJ (2001)
6. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning-1. Information Sciences **8** (1975) 199 – 249
7. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning-2. Information Sciences **8** (1975) 301 – 357
8. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning-3. Information Sciences **9** (1975) 43 – 80
9. Hagras, H.: Type-2 flcs: A new generation of fuzzy controllers. IEEE Computational Intelligence Magazine **2** (2007) 30–44
10. Sepulveda, R., Castillo, O., Melin, P., Rodriguez-Diaz, A., Montiel, O.: Experimental study of intelligent controllers under uncertainty using type-1 and type-2 fuzzy logic. Inf. Sciences **177** (2007) 2023–2048
11. Al-Jaafreh, M., Al-Jumaily, A.: Training type-2 fuzzy system by particle swarm optimization. Evolutionary Computation, 2007. CEC 2007. IEEE Congress on (25-28 Sept. 2007) 3442–3446
12. Sepulveda, R., Castillo, O., Melin, P., Montiel, O., Aguilar, L.: Evolutionary optimization of interval type-2 membership functions using the human evolutionary model. Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International (23-26 July 2007) 1–6
13. Castillo, O., Huesca, G., Valdez, F.: Evolutionary computing for optimizing type-2 fuzzy systems in intelligent control of non-linear dynamic plants. Fuzzy Information Processing Society, 2005. NAFIPS 2005. Annual Meeting of the North American (26-28 June 2005) 247–251
14. Castillo, O., Aguilar, L.T., Cazarez, N., Cardenas, S.: Systematic design of a stable type-2 fuzzy logic controller. Journal of Applied Soft Computing **8** (2008) 1274–1279
15. Castillo, O., Melin, P.: Hybrid Intelligent Systems for Pattern Recognition Using Soft Computing: An Evolutionary Approach for Neural Networks and Fuzzy Systems. Springer-Verlag, Berlin (2005)
16. Cordón, O., Herrera, F., Hoffman, F., Magdalena, L.: Genetic fuzzy systems: Evolutionary tuning and learning of fuzzy knowledge base. World Scientific, Singapore (2001)
17. Lagerberg, A., Egardt, B.: Estimation of backlash with application to automotive powertrains. Proc. of the 42th Conf. on Decision and Control (1999) 135–147
18. Mendel, J., John, R.: Type-2 fuzzy sets made simple. IEEE Transactions on Fuzzy Systems **10** (2002) 117 – 127
19. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. Int. J. Hum.-Comput. Stud. **51** (1999) 135–147
20. Holland, J.M.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor, MI (1975)
21. Aguilar, L.T., Orlov, Y., Cadiou, J.C., Merzouki, R.: Nonlinear H_∞-output regulation of a nonminimum phase servomechanism with backlash. Journal of Dynamic Systems, Measurement, and Control **129** (2007) 544–549

22. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1989)
23. Castro, J.R., Castillo, O., Martinez, L.G.: Interval type-2 fuzzy logic toolbox. Journal of Engineering Letters **15** (2007) 89–98 online version.
24. Castro, J.R., Castillo, O., Melin, P.: An interval type-2 fuzzy logic toolbox for control applications. In: Proc. FUZZ-IEEE 2007. IEEE, London (2007) 61 – 67
25. : Genetic Algorithm and Direct Search Toolbox User's Guide, The Mathworks Inc. Available: http://www.mathworks.com. (2007)