THE USE OF GENETIC ALGORITHMS ON A FUZZY CONTROLLER FOR A SATELLITE ATTITUDE CONTROL DURING THE POINTING PHASE

Cláudio Correa
Sandra Aparecida Sandri
Luiz Carlos Gadelha de Souza
Applied Computing CAP -- Space Mechanics & Controls Division DMC
Brazilian National Institute for Space Research – INPE
CP 515 -- São José dos Campos, 12227-010, Brazil
Phone (0055)(12) 345-6766 -- Fax (0055)(12) 345-6375
E-mail: correa@pgrad.inpe.br, gadelha@dem.inpe.br, sandri@lac.inpe.br

Abstract

In this paper, we propose the use of genetic algorithms (G.A.) for the creation of controllers for the pointing phase of a reaction wheel artificial satellite. We make use of the simulator previously developed which test a PD controller for the pointing phase of a satellite similar to the French-Brazilian Satellite, whose control system is based on the stabilization of 3 axes.

***Key words**:* Genetic Algorithm, Fuzzy Control, Satellite.

Introduction

There exists nowadays a technological tendency of building small artificial satellites, aiming at guaranteeing a fast and simple means of reaching space. The tendency is to have these satellites equipped with highly autonomous systems for attitude, maneuver and orbit control, and to have them developed fastly and at very low costs[1].

In this work, we make use of a model similar to the French-Brazilian[2] satellite one, currently under development at the Brazilian National Institute for Space Research (INPE). This satellite has its control system based on the stabilization of three axes and makes use of a proportional/derivative controller (PD), whose gains have been determined through the poles allocation method, and whose sensor measures have been processed by a Kalman filter[3].

It is important to observe that in contrast to the high complexity of a controller system, it is imperative for it to have the lowest possible cost. For this reason, it is important to investigate different techniques in the development and implementation of a controller. Such innovations can bring versatility in what regards the hardware components to be employed, and in relation to the interfaces among controllers, sensors and actuators.

Fuzzy logic is a one of the most well-succeeded recent technologies in the development of sophisticated control systems[4-5]. Its employment, complex requirements can be implemented in simpler controllers, of easy maintenance and low cost.

The augmentation of satellite autonomy has been long pursued with the goal of not only improvement in performance, but especially in order to reduce fuel consumption. To obtain this autonomy, one approach is the use of fuzzy controllers, especially when the model is subject to uncertainty[6-7-1-8].

A fuzzy controller is composed of a set of rules of the type If <*premise*> then <*conclusion*>, which define control actions in function of some (usually ill-defined) intervals on which the state variables may take their values. These intervals are modeled by fuzzy sets and called fuzzy terms.

The main difficulty in the creation of fuzzy controllers is the definition of the fuzzy terms. One way of dealing with this problem is to use "*neuro-fuzzy*" models[9-10], in which these parameters are learned through the presentation of pairs (input, expected output) to a neural network with nodes that basically compute the intersection and union operations. Another way to learn these parameters is to employ genetic algorithms.

In this paper we present two attitude controllers for the French-Brazilian satellite, built with the use of genetic algorithms. The first one is a PD controller whose gains were found using a genetic algorithm. The second one is a fuzzy controller of the Mamdani type, whose parameters have been learned with another genetic algorithm, using however the same fitness function of the first one.

In the present work, we have used the satellite simulated model, which has been previously developed at INPE using the MATLAB toolkit[3]. We have also taken the PD attitude controller originally developed with the simulator as basis of comparison to assess the quality of the results obtained by the GA based controllers presented here.

This paper is divided as follows. In Section II we present some fundaments about genetic algorithms and fuzzy controllers. In section III we present the model of

the satellite used in our applications, and the original PD attitude controller developed for it in the pointing phase. In Section IV we present the PD and the fuzzy controllers developed using genetic algorithms for the satellite model. Finally, Section V brings the conclusions.

Basic Notions

Genetic Algorithms

Genetic algorithms are adaptive search strategies based on a highly abstract model of biological evolution[11]. They are primarily used in optimization problems for which one aims to find not necessarily an optimal solution, but at least a reasonably good solution.

In these algorithms, a population of individuals (potential solutions) suffers a series of unary transformations (mutation) and of higher order (crossover). These individuals compete among themselves for survival; the most apt individuals have better chances to be chosen to pass their characteristics to the next generation. After some generations, the algorithm (usually) converges and the best individual represents a solution close to the optimum.

The search for the solution involves an evaluation function (fitness), which yields a grade for the performance of each individual, according to aspects considered relevant to the problem at hand. Figure 1 brings an illustration[21] of an evolution cycle in a genetic algorithm.

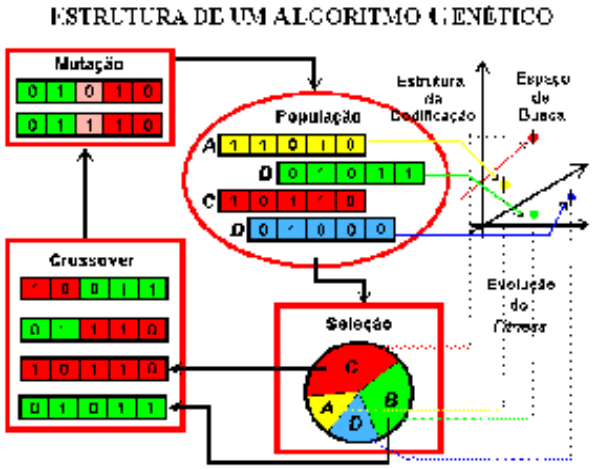

Figure 1: Structure of a simple genetic algorithm.

Genetic algorithms have been used in many applications involving fuzzy control[12-13-14-15-16-17-18], inclusive in Brazil[19-20-21]. In the present work, the fitness function of the G.A. for the pointing phase is a global measure of the performance of each solution in relation to the simulation of a complete orbit under a certain of perturbation.

Fuzzy Controllers

Fuzzy controllers are based on fuzzy sets theory, which has been developed since 1965 after the seminal works of Lotfi Zadeh[22]. Fuzzy control techniques were first developed with the works of E.H. Mamdani[23-24-25], and have been gaining increasing importance over the years, being today the main application of fuzzy sets theory. The term fuzzy logic is usually employed in the control field to name the modeling of fuzzy pieces of information and the inference mechanisms that act upon them.

Contrary to what happens in conventional control in which the control algorithm is described analytically by algebraic or differential equations, by means of a mathematical model, in fuzzy control logical rules are employed in the control algorithm, obtained through the synthetization of human experience, intuition and heuristics, in process control[22].

A rule in a fuzzy controller is usually of the type *If* $x_1 = A_1$ *and* $x_2 = A_2$ *... and* $x_n = A_n$ *then* $y = B$, where the x_i and y are respectively state and control linguistic variables, and the A_i*'s* and B are linguistic terms. A linguistic variable is a 4-tuple $(x, T(x), \Omega, M)$, where x is the name of the variable, Ω is the domain of x, $T(x)$ is a set of linguistic terms, i.e. a set of names of fuzzy sets, and M is a function that associates a fuzzy set in Ω to each term in $T(x)$. Figure 2 brings illustrates the linguistic variable "*error*" with terms $T(error)=\{negative\_big, negative\_small, zero, positive\_small, positive\_big\}$. A rule in a fuzzy controller could be for instance be "*If error = small then throttle = big*".

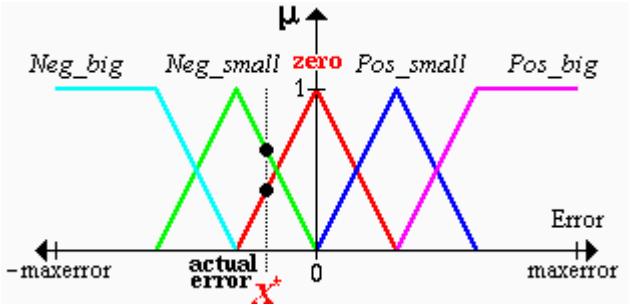

Figure 2: Linguistic terms of variable "throttle".

The basic structure of a fuzzy controller is illustrated[5] in Figure 3. The main components are the knowledge base that contains the rules and the description of the linguistic variables, and the inference engine, that

allows us to obtain a control action in function of the value of the state variables in a given moment of time.

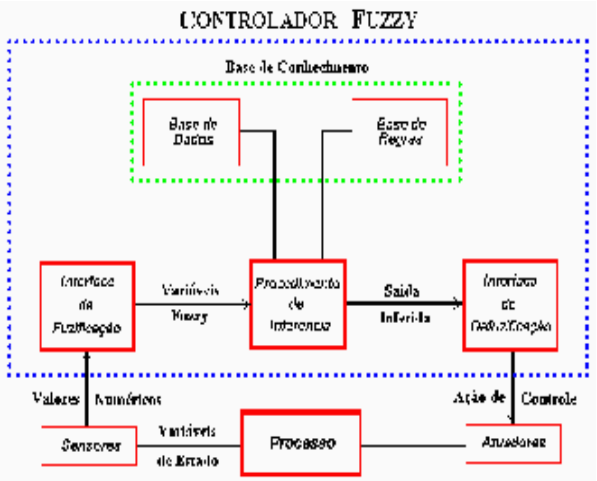

Figure 3: Structure of a fuzzy controller.

Fuzzy controllers are highly adaptable and capable of incorporating knowledge that many other systems are incapable of doing[27]. They are also versatile, especially when the physical model is very complex and of difficult mathematical reproduction.

In general, they are more useful in non-linear systems, varying in time or not, support very well perturbations and highly noisy plants, and are robust even in systems where uncertainty is intrinsically present.

Satellite Model and Original PD Controller

Satellite Model

The French-Brazilian satellite model is a rigid body model where the null inertia product is neglected. The control system consisted of a gyro and a star and sun sensors plus a PD controller in the pointing phase. The rotational dynamics of the satellite can be represented by the following differential equations system[3]:

$$\begin{aligned} \dot{\xi} &= \varpi \\ \dot{\varpi} &= -K_P * \hat{\xi} - K_D * \hat{\varpi} + T_P(t) \\ \dot{\hat{\xi}} &= \hat{\varpi} \end{aligned} \qquad (1)$$

where $\hat{\xi}$, $\hat{\varpi}$ and $T_P(t)$ represent the respective estimation of rotation (ξ), angular speed (ϖ) and perturbation torque's. The estimation of the angular speed (ϖ) is determined with:

$$\hat{\varpi} = o - \hat{b}_k \qquad (2)$$

and the gyro output (o) is given by:

$$o(t) = \varpi + b_k + n_k; \qquad t \in [t_k; t_{k+1}) \qquad (3)$$

where b_k represents the long period bias (derive) and n_k the short period bias (noise).

From Eq. 2 and Eq. 3 we have:

$$\hat{\varpi} = \varpi + b_k - \hat{b}_k + \sigma_n^2 \qquad (4)$$

where σ is the covariance of noise n.

The long period bias and its estimation can be propagated in time with:

$$\begin{aligned} b_{k+1} &= e^{-\lambda * \Delta t} * b_k + n(2 * \lambda * \Delta t * \sigma_d^2 \\ \hat{b}_{k+1} &= e^{-\lambda * \Delta t} * \hat{b}_k \end{aligned} \qquad (5)$$

where λ represents the correlation and σ_d the long period covariance.

Making $D_k = b_k - \hat{b}_k + n(\sigma_n^2)$, Eq. 1 becomes:

$$\begin{aligned} \dot{\xi} &= \varpi \\ \dot{\varpi} &= -K_P * \hat{\xi} - K_D * (\varpi + D_k) + T_P(t) \\ \dot{\hat{\xi}} &= (\varpi + D_k) \end{aligned} \qquad (6)$$

The coordinate system for the satellite attitude control adopts an external referential, defined as:

- Z_E oriented to the north pole of the ecliptic;
- X_E pointing to the sun position and
- Y_E pointing in such a way as to form a destrogerous system.

The simulations made with a PD controller used the following parameters[3]:

1. Pointing precision:

- 0.5° for axis x;
- 0.15° for axes y and z.

2. Stability:
   - 0.05°/s for the 3 axes.

The following orbital data has been used:

- Altitude at apogee 1500 km;
- Altitude at perigee 400 km;
- Inclination of 7°;
- Argument at perigee of 90°;
- Longitude of the descending node of -90° and
- Mean anomaly of -90°.

Original PD Controller

In the present work, we have used the satellite simulated model, which has been previously developed at INPE using the MATLAB toolkit[3]. We have also taken the PD attitude controller originally developed with the simulator as basis of comparison to assess the quality of the results obtained by the GA based controllers presented here.

In the following we present the he original attitude controller for the pointing phase. It consists of a PD controller for each axis, using the following equation:

$$f(t) = \mathrm{K_P} * e(t) + \mathrm{K_D} * \Delta e(t) \quad \textbf{(7)}$$

where $e(t)$ is the signal error and $\Delta e(t) = \frac{d}{dt}[e(t)]$, $\mathrm{K_P}$ and $\mathrm{K_D}$ are the proportional and derivative gains.

In order to determine the gains of this controller the Pole allocation method has been employed, which through project restrictions such as peak time (T_P) and accommodation time (T_S), allows us to obtain the position state vector (ξ) and the angular speed (ϖ).

The gains have then been calculated as a function of these specifications, which become implicit in the state vector. Equal gains (proportional and the derivative) have been used for X and Y axis, in order to simplify calculations.

The PD controller responds proportionally to the angular position error $(\zeta(t))$ and its derivative error, which is equivalent to the angular speed $(\Delta\zeta(t))$. The tuning has been performed through the selection of the gains which lead to a satisfactory answer, i.e. the control function ought to maintain the controlled variables as close as possible to the desired values.

The satellite dynamical equations consider the perturbations due to 3 torque's: atmospheric drag, solar radiation pressure and magnetic field. These torque's originate the errors at the satellite pointing, and the function of the controller is to react to the perturbations sending a signal to the atuactors, which then generate an action to correct the pointing along the orbital trajectory.

The gains found in the original model are:

- Proportional gains for the *X* and *Y* axes, $\mathrm{K_t} = 0.0263$;
- Derivative gains for the *X* and *Y* axes, $\mathrm{K_t} = 0.08$;
- Proportional gain for the *Z*, $\mathrm{K_{tz}} = 0.0272$;
- Derivative gain for the *Z*, $\mathrm{K_{wz}} = 0.1$.

G.A. Built Attitude Controllers for the Satellite Pointing Phase

In the following we present the G.A. built attitude controllers for the satellite-pointing phase. We first present the simulation conditions used in both developments, to then present the PD and the fuzzy attitude controllers developed using G.A's.

Simulation Conditions

In order to effectively assess the quality of the controllers built with the G.A.'s. severe conditions have been adopted for the satellite operation mode. In this sense, the magnitude order of the perturbations has been incremented in relation to those adopted to assess the quality of the original PD controller[3]. The perturbations due to atmospheric drag and solar radiation pressure have been multiplied by constants $\beta_a = 10$ and $\beta_s = 100$ respectively, corresponding to the value necessary to reach the same order of magnitude of the perturbations due to the magnetic field, which had so far a predominant role over the other values (see Fig. 4).

The satellite state vectors at the initial moment represents a zero error condition, in what regards both pointing and speed. In the present work, an error close to the limits imposed by the project specifications has been adopted for the position of the 3 axes, with speed kept as in the original situation. The initial values for the 3 axes are: $X = 0.35^\circ$, $Y = -0.12^\circ$ and $Z = 0.12^\circ$.

The attitude error relative to the control actions in the simulation of an orbit are shown in Fig. 5 and Fig. 6, calculated as the original control laws[3]. To evaluate robustness of the G.A. built controllers, we compare

their results with the ones obtained by the original PD controller.

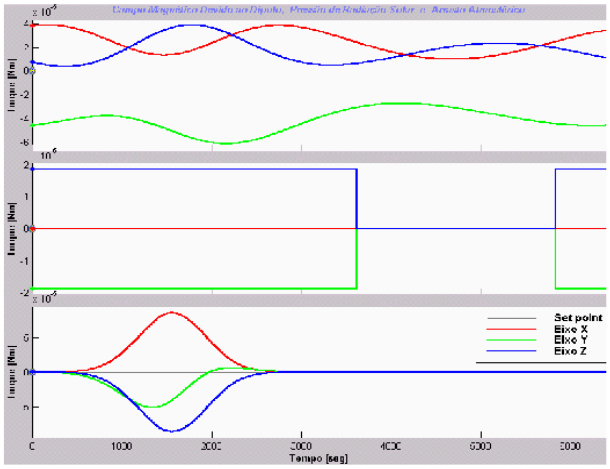

Figure 4: Perturbation torque's multiplied by β.

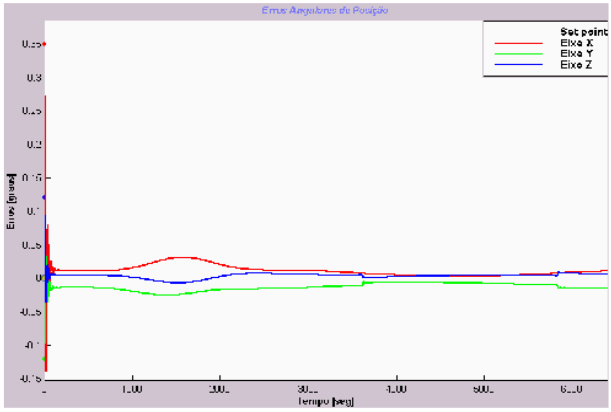

Figure 5: Reference position errors for the simulations.

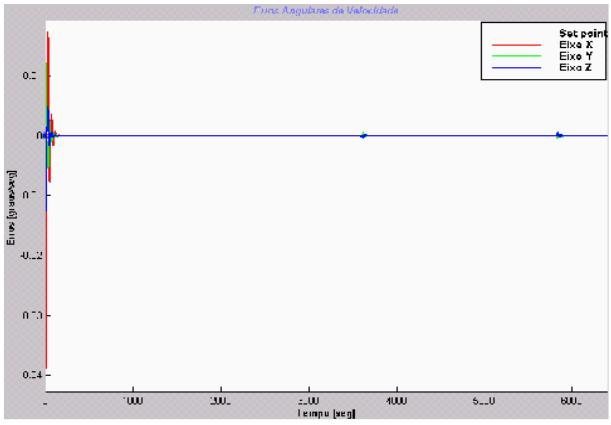

Figure 6: Reference speed errors for the simulations.

G.A. Characteristics

A GAS has been used to built the controllers for the satellite model in this work. Each chromosome in a given population is composed of a set of parameters specifying a controller. In the case of the PD controller, the chromosomes will contain the proportional and derivative gains, and in the case of the fuzzy controller, they will contain the fuzzy terms employed by the rules.

The fitness function of the GAS employed to learn the controller parameters determines a measure on the performance of the candidate controllers, each of which built using the parameters specified in its respective chromosome. This function needs to take into account the relevant aspects in the controller answer, in order to guarantee performance and stability.

Since the fitness function depends only on the results of the simulation of each candidate controller, the same fitness function can be used to optimize different types of controllers. In particular, in this work, a single fitness function has been employed to obtain optimized PD and fuzzy controller.

In order to simplify the evaluation, only variables (ζ, ω) are used in the function. Moreover, all candidate solutions violating a project specification are eliminated.

1. Position error restrictions (ζ):
   - Eliminate chromosomes for which $\zeta_X > 0.5^\circ$, or $\zeta_Y > 0.15^\circ$, or $\zeta_Z > 0.15^\circ$;
2. Angular speed error restrictions (ω):
   - Eliminate chromosomes for which $\omega_X > 0.05^\circ / s$ or $\omega_Y > 0.05^\circ / s$, or $\omega_Z > 0.05^\circ / s$.

The performance of the GAS was studied using different fitness functions. The best results were obtained using an approximation of the angular position error. This is done by function *trapz.m*, available in Matlab, which calculates a numerical integral by the trapezoidal sum method. An abbreviated version of the fitness function is given by:

$$fitness = \left[1 - \frac{\sum_{i=1}^{n} \left(\int_0^{t_f} f(\theta) d\theta \right)_i * \nu_i}{0.15 * n} \right] \quad \textbf{(8)}$$

where $f(\theta)$ represents the position error for each of the three, ν_i corresponds to weights for each axis, is the number of variables (axes) and t_f is the time taken by the simulated orbits.

In all the tests, the same parameters for the GAS have been maintained: population composed of 30 individuals, $p_c = 0.9$, $p_m = 0.033$, sensibility of 3 decimal cases to detect fitness variation and as last stop criteria a maximum number of 35 generations. Each chromosome is represented by a bit vector, with each gene (parameter) occupying 20 bits. The total size of a chromosome depends on the number of codified parameters.

Each chromosome has been evaluated in relation to the simulation of 1/4 of an orbit, which correspond to 1600 seconds.

G.A. Built Controllers

We have used the simulation conditions and the GAS structure described above to generate the gains of a PD controller, with a control law for each axis. Therefore, 8 gains were codified in each chromosome. The GAS converged after 23 generations; the results of this controller are shown in Figure 7.

With the same GAS parameters specification and fitness function given above, we have developed fuzzy controllers of the Mamdani type, using the fuzzy control toolbox available in Matlab.

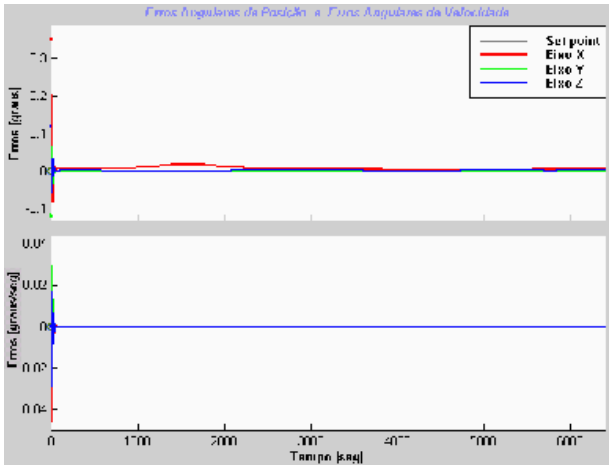

Figure 7: Errors obtained with the use of a PD controller optimized by a GAS.

In this case, the number of fuzzy rules is fixed, as well was the identity of the terms appearing on each rule. The rules were built through the observation of the PD controller. The chromosomes then encode the centers of the fuzzy terms appearing in the premises and conclusion of the rules.

All the fuzzy sets are triangular, except those on the extremities, which are trapezoidal in shape. All fuzzy sets are symmetrical in relation to 0, and therefore, only the positive centers have to be learned. To confer more smoothness to the control surface, neighboring the fuzzy sets are superposed. These restrictions do not limit the model, and have been frequently used in fuzzy controllers [4-28-29-30].

To further reduce the number of parameters to be learned we have used a single control for all the axes. The surface of best Mamdani fuzzy controller obtained, with 3 fuzzy terms for each input variable and 5 fuzzy terms the output variable (torque), is depicted in Figure 8. The results of this controller are shown in Figure 9.

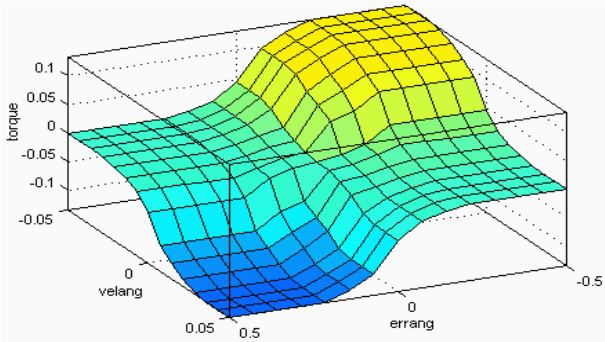

Figure 8: Mapping of the output torque of the fuzzy controller.

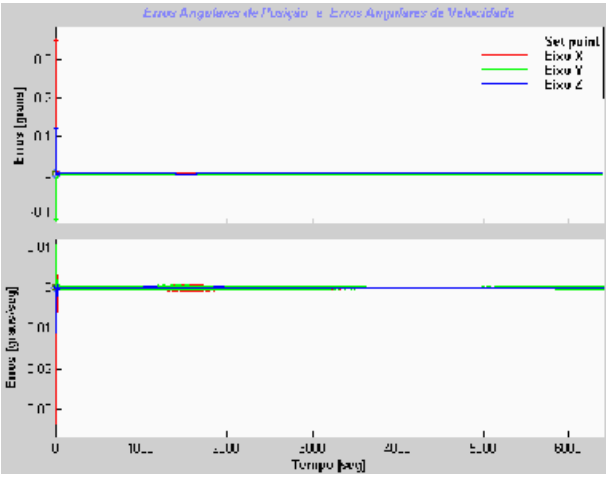

Figure 9: Errors obtained with the use of a fuzzy controller optimized by a GAS.

Conclusions

We have presented a PD and a fuzzy controller for attitude control of a satellite model, whose parameters have been obtained through the use of a simple genetic algorithm. Table 1 below brings a comparison of the performance of these controllers in terms of the sum of the errors yielded by each controller during the simulation of a single orbit. The table also brings the performance of the original PD controller developed for the satellite model using the poles allocation method. It is easy to see that the pointing error is decreasing in PD, PD-GA and CN-GA controller sequence. However, this not happens in the velocity errors as for the CN-GA controller, although, it keeps the velocity errors under specifications design.

Table 1: Results of the integral of the errors and total sum of the integral in relation to the 3 axes.

Absolute value of the integral *(Trapezoidal method)*				
Controller	***Angular Errors***			
	Positions			
	Axis X	Axis Y	Axis Z	*Total*
PD	78.1237	84.5526	31.1589	193.8354
PD-GA	43.1680	12.6005	7.1152	62.8839
CN-GA	5.3271	4.1397	2.0544	11.5214
Controller	Speed			
	Axis X	Axis Y	Axis Z	*Total*
PD	0.93837	0.32763	0.29725	1.56326
PD-GA	0.61499	0.44636	0.38184	1.44321
CN-GA	1.42073	1.31787	0.56772	3.30633

The controllers obtained are robust, and the results produced validate the use of genetic algorithms as an optimization tool to treat similar problems as those addressed in this work.

References

[1]Guerra, R.; Sandri, S. A.; Souza, M. L. O. Dynamics and design of autonomous attitude control of a satellite using fuzzy logic. *Anais do Congresso Brasileiro de Engenharia Mecânica*, Bauru, Brazil, n. COB 1338, 1997. (COBEM'97).

[2]Souza, L.C.G.and Silva,A.R., ,"French-Brazilian Micro-Satellite Control System Design During Normal Mode". In Applied Mechanics in the Americas, Vol, 8, Ed. H. I. Webwer, P.B. Gonçalves, I. Jasiuk, D. Pamplona, C. Steele and L. Bevilacqua, pp. 1163-1166, Published by AAM and ABCM – Rio de Janeiro, 1999. ISBN: 85-900726-2-2.

[3]Silva, A. R. da Estudo de sistema de controle de um satélite artificial durante a fase de transferência orbital e apontamento. *Dissertação de Mestrado em Eng. Espacial e Tec. Espaciais/Mecânica Espacial e Controle*. Instituto Nacional de Pesquisas Espaciais, São José dos Campos, Brasil, INPE-6397-TDI/613, 86 p., 1997.

[4]Driankov, D.; Hellendoorn, H.; Reinfrank, M. *An Introduction to Fuzzy Control*. Springer-Verlag, 1993.

[5]Lee, C. C. Fuzzy logic in control systems: Fuzzy logic controller (part I). *IEEE Transactions on Systems, Man an Cybernetics*, v. 20, n. 2, p. 404-418, 1990.

[6]Woodward, M. A. Fuzzy open-loop attitude control for the FAST spacecraft. *Proc. of the NASA AIAA, Guidance, Navigation and Control Conference*, San Diego, July 1996.

[7]Conway, D.; Sperling, R.; Folta, D.; Richon, K.; DeFazio, R. Automated maneuver planning using a fuzzy logic controller. *Proc. of the NASA AIAA, Guidance, Navigation and Control Conference*, San Diego, 1994.

[8]Guerra, R.; Sandri, S. A.; Souza, M. L. O. Controle de atitude autônomo de satélites usando lógica nebulosa. *III Simpósio Brasileiro de Automação Inteligente*, Vitória, Brazil, p. 337-342, 1997. (SBAI'97).

[9]Jang, J. S. R. ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man an Cyberneticss*, v. 23, n. 3, p. 665-685, 1993.

[10]Lin, C. -T. A neural fuzzy control system with structure and parameter learning. *Fuzzy Sets and Systems*, n. 70, p. 183-212, 1995.

[11]Husbands, P. Genetic algorithms in optimization and adaptation. In: Kronsjo, L.; Shunsheruddin, D., ed. *Advances in paralell algorithms*. Blackwell Scientific Publications, p. 227-277, 1992.

[12]Karr, C. Applying genetics to fuzzy logic. *AI Expert*, p. 38-43, 1991.

[13]Karr, C. Adaptive control with fuzzy logic and genetic algorithms. In: Yager, R. R.; Zadeh, L. A., ed. *Fuzzy sets, neural networks and soft computing*. Van Nostrand Reinhold, p. 345 – 367, 1994.

[14]Pham, D. T.; Karaboga, D. Optimum design of fuzzy logic controllers using genetic algorithms. *J. Syst. Eng*., v. 1, p. 114-118, 1991.

[15]Kinzel, J.; Klawonn, F.; Kruse, R. Modifications of genetic algorithms for designing and optimizing fuzzy controlers. *Proc. of the IEEE I Int. Conf. on Evolutionary Computation*, Orlando, p. 28-33, 1994.

[16]Kim, J.; Moon, Y.; Zeigler, B. P. Designing Fuzzy Net Controllers Using GA Optimization. *Proc. of IEEE/IFAC Joint Symposium on Computer-Aided Control System Design. Tucson*, 1994.

[17]Mohamadian, M.; Stonier, R. J. Generating fuzzy rules by genetic algorithms. *Proc. of the IEEE III Int.*

Workshop on Robot and Human Communication, Nagoya, p. 362-367, 1994.
[18]Mohamadian, M.; Stonier, R. J. Tuning and optimization of membership of fuzzy logic controlers by genetic algorithms. *Proc. of the IEEE III Int. Workshop on Robot and Human Communication*, Nagoya, p. 356-361, 1994.
[19]Aya, J. C. C.; Costa, O. L. V. Otimização de controladores nebulosos usando algoritmos genéticos. *III Simpósio Brasileiro de Automação Inteligente*, Vitória, p 207-212, 1997. (SBAI'97).
[20]Silva, J. C. A.; Kierner, C. Algoritmo genético aplicado a simulação. *III Simp\'osio Brasileiro de Automação Inteligente*, Vitória, p. 283- 288, 1997. (SBAI'97).
[21]Vasconcelos, J. A. D.; Neiba, A. M.; Cardoso, E. N.; Pinheiro, F. A. Algoritmo genético aplicado ao controle de tensão em sistemas elétricos de potência. *III Simpósio Brasileiro de Automação Inteligente*, Vitória, p. 38-44, 1997.
[21]Hoffmann, F. Genetic algorithm. [online] <http://www.cs.berkeley.edu:80/~fhoffman/ga.html>, May 1997.
[22]Zadeh, L. A. Fuzzy sets, information and control, 1965. v. 8. p. 338-353.
[23]Mamdani, E. H. Advances in the linguistic syntesis of fuzzy controllers. *Int. J. Man-Mach. Stud*, v. 8, p. 669-678, 1976.
[24]Mamdani, E. H.; Baaklini, N. Prescriptive method for deriving control policy in a fuzzy logic controller. *Eletronic Letters*, v. 11, p. 625-626, 1975.
[25]Mamdani, E. H.; Procyk, T.; Baaklini, N. Application of fuzzy logic to controller design based on linguistic protocol. In: Mamdani, E. H.; Gaines, B. R., ed. *Discrete systems and fuzzy reasoning*. London: University of London. p. 125 - 149.
[26]Bauer, P.; Nouak, S.; Winkler, R. A brief course in fuzzy logic and fuzzy control. [online]. <ftp://ftp.flll.uni-linz.ac.at/pub/info/flllorg.ps.gz>. Apr. 1998.
[27]Guerra, R. Projeto e simulação do controle de atitude autônomo de satélites usando lógica nebulosa. *Dissertação de Mestrado em Eng. Espacial e Tec. Espaciais/Mecânica Espacial e Controle*. Instituto Nacional de Pesquisas Espaciais, São José dos Campos, INPE-6714-TDI/630, 190 p., 1998.
[28]Heider H., Tryba V., Mühlenfeld E. Automatic design of fuzzy systems by genetic algorithms. In: Bouchon-Meunier B., Yager R. R., Zadeh L. A., ed. *Fuzzy Logic and Soft Computing*. World Scientific, p. 21-28, 1995.
[29]Larsen, P. M. Industrial aplications of fuzzy logic control. In: Mamdani, E. H.; Gaines, B. R., ed. *Fuzzy Reasoning and its Aplications.* London: Academic Press Inc., p. 335 – 342, 1981.
[30]Sanchez, E.; Shibata, T.; Zadeh, L. A. *Genetic algoritms and fuzzy logic systems: soft computing perspectives*. Singapore: World Scientific Publishing Co. Pte. Ltd., 240 p., 1997.