Optimization of Fuzzy Attitude Control for Nanosatellites

Ástor del Castañedo[1], Daniel Calvo[2], Álvaro Bello[2], and María Victoria Lapuerta[3(✉)]

[1] Universidad Politécnica de Madrid, Madrid, Spain
[2] Universidad Politécnica de Madrid, E-USOC, Madrid, Spain
[3] Universidad Politécnica de Madrid, ETSIAE, Madrid, Spain
mariavictoria.lapuerta@upm.es

Abstract. In this work, the optimization by genetic algorithms of a system based on fuzzy logic for the attitude control of a nanosatellite is performed. The objective of this optimization is to propose different designs of the fuzzy controller depending on the possible operation modes along the whole mission of the satellite, to improve its efficiency and performance. Both, mono and multi objective optimizations, are performed finding that mono objective optimization leads to results that are not applicable in real systems due to its high cost of electrical power and that multi objective optimizations give very interesting results which allow some flexibility to change the controller to a faster one or one of lower cost.

Keywords: Fuzzy logic · Genetic algorithms · Attitude control

1 Introduction

The Attitude Determination and Control Subsystem (ADCS) is critical for the majority of space missions. The main requirements for this subsystem are: (i) Accuracy: Most space missions have attitude requirements to operate their payloads that can range from a few degrees of pointing accuracy up to a fraction of arcsecond for some scientific missions. (ii) Low power consumption: This is the most important requirement for small satellite missions, as usually the budget is very limited. The less energy is required for attitude control the more energy is available for other subsystems and for the payload. (iii) Stability and robustness: An unstable attitude control may lead to a mission failure [1].

Many different controllers have been used or studied for space applications. From traditional satellite control [2] which has relied on classic control theories such as Proportional Integral and Derivative (PID) [3], or Linear Quadratic Regulator (LQR) controllers [4], to intelligent control based on Fuzzy logic [5–10] or learning algorithms like neural networks [11, 12].

Fuzzy logic is a mature control theory [13–16] already used in many commercial applications [17–19]. It is characterized by simplifying the design process by taking advantage of the knowledge of selected experts [17, 18, 20]. So, taking into account

K. Arai et al. (Eds.): IntelliSys 2018, AISC 869, pp. 970–990, 2019.
https://doi.org/10.1007/978-3-030-01057-7_72

that satellite designers usually have extensive experience in satellite environment and dynamics, the use of fuzzy logic for control could be highly suitable for small satellites.

However, although fuzzy logic has shown good results when its application to space missions has been studied [5–10], it has experienced a tepid adoption in global scale mainly because, in the field of the space technologies, changes are introduced in a slow and careful way. So, it is necessary not only to develop and implement intelligent controllers but also to compare and test their performance and efficiency with the traditional controllers through simulations and on board demonstration.

In Walker's work [9], a Linear Quadratic Regulator (LQR) control performance was compared with a fuzzy controller in a CubeSat through simulations, finding that the fuzzy controller was a lower-cost solution than the LQR and also tends to settle faster than the LQR.

In Calvo's work [21], a tailored Adaptive Fuzzy controller was designed for a nanosatellite mission. Its performance and efficiency were compared, through numerical simulations, for the same specific mission, with a traditional PID controller, which is the classic control theory, most used and tested in ADCS of satellites. Both controllers were designed in order to be tested on board a satellite called QBITO, as part of the QB50 mission. The main objective of QB50 was the study of the composition of the lower thermosphere by launching 50 nanosatellites in the same mission (for more details, see [21]). The numerical comparison between both controllers, Fuzzy and PID, showed that the fuzzy controller was much more efficient, in terms of power consumption, and also achieving better precision in general than the PID.

The fuzzy controller developed in [21] was designed to achieve low energy consumption, as the available power of the mission was strongly constrained. However, depending on the mission requirements and the external perturbations, different operation modes could be necessary along the whole mission. Those operation modes are the answer to a variety of needs by getting different combinations of power consumption and accuracy. This work is based on the use and optimization of the fuzzy logic model developed in [21], to improve its efficiency and performance and to propose different designs of the fuzzy controller depending on the possible situations during the mission.

Nowadays, there are multiple optimization systems. The choice between them is usually based on the form of the objective function within the solution space. In this case, as the form of the objective function in unknown, one heuristic optimization method is used: the so-called genetic algorithms (GA), which, although their computational cost is much higher than the other methods, ensures that the solution obtained is a good minimum.

Therefore, the objective of this work is the optimization by GA of the attitude control system based on fuzzy logic of [21] and its comparison with the system designed there. The same simulator and data used in the design of the control system are used in the optimization to make the comparison as realistic as possible.

This paper is organized as follows: In Sect. 2 a brief mission overview is depicted emphasizing the constraints and the requirements to be accomplished. In Sect. 3 an overview of the attitude control is described, being the Fuzzy controller of [21] described in Sect. 4. In Sect. 5 monoobjective optimizations are performed and in

Sect. 6 multiobjective optimizations are performed. Finally, in Sect. 7, the conclusions of this work are stated and discussed.

2 Mission Overview

In this section, the details of the QB50 mission for which the fuzzy controller was designed are described. We are going to use the same mission in this work for the optimization of the controller.

The QB50 mission was a project of the 7th framework program lead by the Von Karman Institute (VKI). The main objective of this project was to study the composition of the lower thermosphere by launching together 50 nanosatellites distributed in a 'string-of-pearls' configuration. Our research group developed one of these satellites, called QBITO.

The starting point of the mission was a circular orbit at 380 km of altitude and $i = 98°$ of inclination (see Fig. 1 where the orbit frame is also represented). Due to atmospheric drag, the satellite orbits decay until the spacecraft burns in the atmosphere. The main payload of the satellites was an Ion Neutral Mass Spectrometer (INMS). The INMS imposed some requirements on the mission which applied directly to the ADCS. In particular, in the nominal attitude the body reference system axes (see Fig. 2) should coincide with the orbit reference system one. When the scientific unit would be acquiring data, it would be necessary to have a pointing accuracy of less

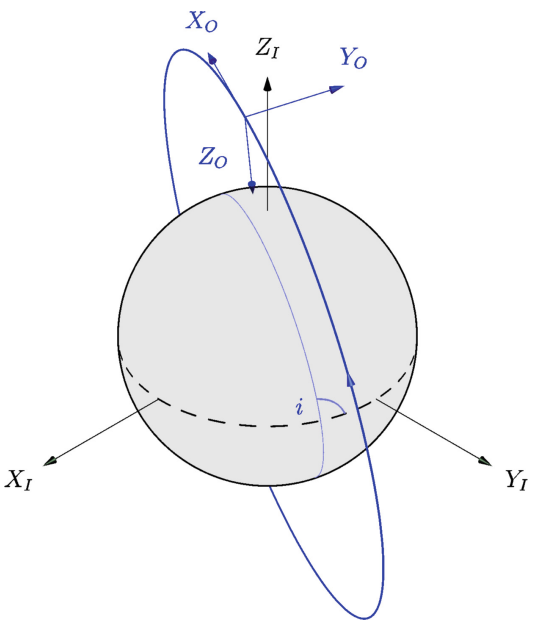

Fig. 1. QBITO inclined orbit representation along with the Orbit (Xo, Yo, Zo) and Inertial (X_I, Y_I, Z_I) Reference Frames (not in scale). i represents the orbit inclination.

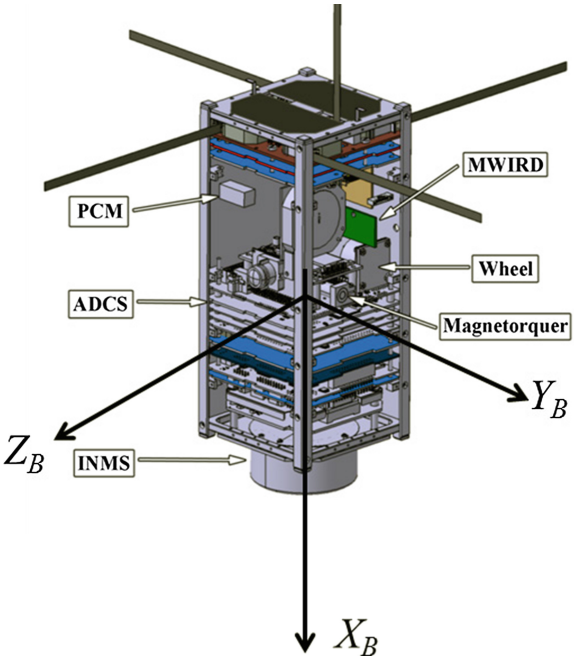

Fig. 2. Open view of QBITO with all the payloads and ADCS actuators labelled… (X_B, Y_B, Z_B) is the body fixed reference frame.

than $\pm 10°$ from the ram velocity vector. In addition, the actual attitude of the satellite should be known with less than $\pm 2°$ accuracy. Both requirements should be achievable until a height of 200 km would be reached, which implies that significant disturbance torques would have to be counteracted in order to comply with them.

3 Attitude Controller

Since it is not possible to use the real satellite in orbit as a plant in the design process, a tailored Dynamics, Kinematics and Environment simulator (DKE) was implemented to be used as the plant to be controlled (see Fig. 3). This figure shows the control scheme with the typical elements: the plant (DKE), sensors, actuators and the controller. QBITO had two sets of actuators, three magnetorquers, one per axis; and one momentum wheel mounted with its axis parallel to the body Y axis (Y_B). Magnetorquer for Y axis is only used when the wheel is saturated. As this study is focused on the controller, the sensors have not been modeled, thus the attitude is known by the controller with no error, and the actuators are modeled only including a saturation in their actuation.

The core of the DKE is the equations of motion of a spacecraft with its momentum exchange devices, see (1), as in [3]. The DKE simulator also includes a space environment model and the disturbance torques affecting the satellite dynamics. It also

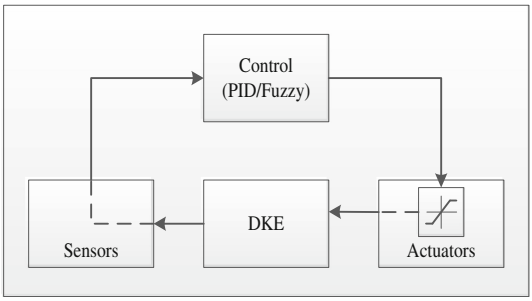

Fig. 3. Control schema.

provides Sun ephemeris which is used in the satellite attitude control software but is not used in this paper since no attitude determination is performed. Finally, the main mission of the DKE is to propagate the satellite dynamic state (attitude and spatial position) taking into account the control actuations and the disturbance torques.

$$d\boldsymbol{\Omega}/dt = \boldsymbol{I}^{-1}\left(\boldsymbol{T} + \boldsymbol{I}\boldsymbol{\Omega} \times \boldsymbol{\Omega} - \boldsymbol{I}_w \frac{d\boldsymbol{\omega}_w}{dt} + \boldsymbol{I}_w\boldsymbol{\omega}_w \times \boldsymbol{\Omega}\right) \tag{1}$$

In (1) $\boldsymbol{\Omega}$ is the spacecraft (body reference frame) angular velocity w.r.t. the inertial reference frame, expressed in body reference frame. $\boldsymbol{T}$ are the external torques, both disturbance (e.g. aerodynamic drag torque) and control (e.g. magnetorquer actuation). $\boldsymbol{\omega_w}$ is the momentum wheel rotational speed expressed in body reference frame. $\boldsymbol{I}$ and $\boldsymbol{I_w}$ are the spacecraft and momentum wheel inertia tensors, respectively, measured in the body reference frame.

Quaternions are used for the attitude representation:

$$\boldsymbol{q} = \begin{pmatrix} \boldsymbol{v}(1)\sin\left(\frac{\delta}{2}\right) & \boldsymbol{v}(2)\sin\left(\frac{\delta}{2}\right) & \boldsymbol{v}(3)\sin\left(\frac{\delta}{2}\right) & \cos\left(\frac{\delta}{2}\right) \end{pmatrix} \tag{2}$$

where $\boldsymbol{v}$ represents the Euler axis unit vector and δ is the angle of the rotation, see Fig. 4, which describes any possible rotation avoiding singularities.

For the control laws next equations are used:

$$\boldsymbol{q_E} = \boldsymbol{q_S}^{-1}\boldsymbol{q_T} \tag{3}$$

$$\boldsymbol{\Omega_{OB}} = \begin{Bmatrix} \Omega_{OB_x} & \Omega_{OB_y} & \Omega_{OB_z} \end{Bmatrix} \tag{4}$$

where $\boldsymbol{q_E}$, $\boldsymbol{q_S}$ and $\boldsymbol{q_T}$ are respectively the Error, Spacecraft and Target quaternions and $\boldsymbol{\Omega_{OB}}$ represents the angular velocity vector of the orbital reference frame w.r.t. the body reference frame, with its components $\boldsymbol{\Omega_{OBx}}$, $\boldsymbol{\Omega_{OBy}}$ and $\boldsymbol{\Omega_{OBz}}$. Unless it is specified, all the vectors and axes referred to in this paper are expressed in the body reference frame.

From the quaternions in (3) three angles are defined, which are used in the control laws:

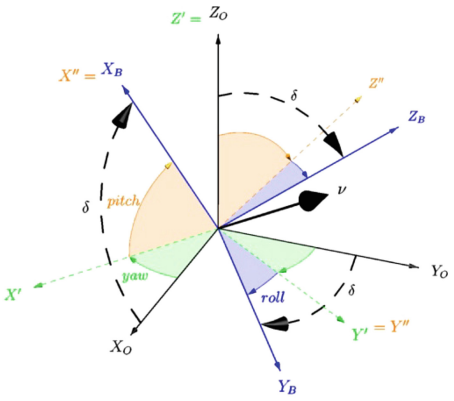

Fig. 4. Orbital Reference Frame and Body Reference Frame for an arbitrary attitude. (X', Y', Z') and (X'', Y'', Z'') are intermediate frames with no physical meaning. v and δ are the Euler axis unit vector and angle representation of the rotation.

$$\begin{aligned} \gamma &= \boldsymbol{q_E}(1)\boldsymbol{q_E}(4) \\ \sigma &= \boldsymbol{q_E}(2)\boldsymbol{q_E}(4) \\ \mu &= \boldsymbol{q_E}(3)\boldsymbol{q_E}(4) \end{aligned} \tag{5}$$

Finally, the relationship between quaternions and the Euler angles is given by;

$$\begin{aligned} \text{roll} &= \text{atan}\left(\frac{2(\boldsymbol{q_S}(2)\boldsymbol{q_S}(3)+\boldsymbol{q_S}(1)\boldsymbol{q_S}(4))}{1-2\left(\boldsymbol{q_S}(1)^2+\boldsymbol{q_S}(2)^2\right)}\right) \\ \text{pitch} &= \text{asin}(2(\boldsymbol{q_S}(4)\boldsymbol{q_S}(2)-\boldsymbol{q_S}(3)\boldsymbol{q_S}(1))) \\ \text{yaw} &= \text{atan}\left(\frac{2(\boldsymbol{q_S}(2)\boldsymbol{q_S}(1)+\boldsymbol{q_S}(3)\boldsymbol{q_S}(4))}{1-2\left(\boldsymbol{q_S}(3)^2+\boldsymbol{q_S}(2)^2\right)}\right) \end{aligned} \tag{6}$$

4 Fuzzy Controller

Fuzzy controller was designed based on the guidelines on [15]. These recommend starting with a fuzzified PD controller. In addition, since the dynamics and the actuators available in the three axes were different, independent controllers were designed for each of them. A detailed schema of the Fuzzy controller is depicted in Fig. 5. The difference between the attitude of the satellite and the desired one produces an error that enters into the control system and is multiplied by the corresponding gain and evaluated in the fuzzy control block. From this block a control torque is obtained, which is multiplied by another gain. This torque will be the one that realizes the magnetorquer of the corresponding axis, reason why saturation is applied so that the real values that the magnetorquer is able to provide are not exceed. These control torques are added to the external torques that the satellite suffers and are introduced in the DKE (see Fig. 3)

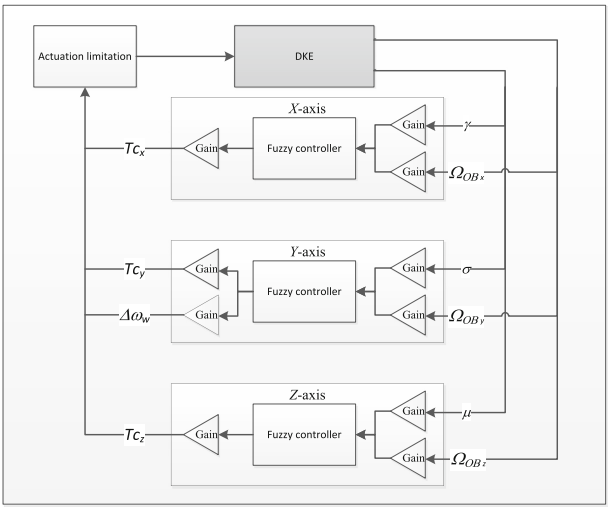

Fig. 5. Fuzzy controller schema.

from where the new attitude of the satellite is obtained and compared again, with the desired one.

Fuzzy controllers are described by their membership functions. The choice of the membership functions was based in the knowledge of experts and in some considerations (see [21]): (i) The first consideration is that the membership functions shall be symmetrical with respect to the central position. (ii) The membership functions selected are triangular for two reasons: (a) the computational cost is very low due to their simplicity and (b) the response is more sensitive to small variations around its center than the one obtained with functions with null derivative at the central position. (iii) The membership functions have been designed in layers around the central one, allowing different responses depending on the state of the system. (iv) In addition, the lateral membership functions are asymmetrical, skewed to the center, producing a more abrupt response when the system is closer to the nominal position than when it is farther away from it. Nevertheless, to allow smooth transitions between membership functions, the overlapping has been set at 50% in all cases. Finally, taking into account those considerations as well as the acquired knowledge of the system, the membership functions have been iteratively modified until good behavior has been achieved Here we show, as an example, the membership functions for σ angle and the orbital angular velocity $\left(\boldsymbol{\Omega}_{\boldsymbol{OBy}}\right)$ (see Figs. 6 and 7, respectively).

The de-fuzzification Sugeno method was used for all the fuzzy controllers. Therefore, there are no output membership functions, just a list of discrete values which represent the actuation fraction to be used by either T_{cj} (magnetorquers) or $\Delta\omega_w$ (momentum wheel). The rules of the controllers were detailed in [21].

Figure 5 shows that a set of gains is needed to set the operating point per controller. The values of the gains were obtained by performing a calibration process, in which each axis was tuned separately. The first calibration was made using the Ziegler-

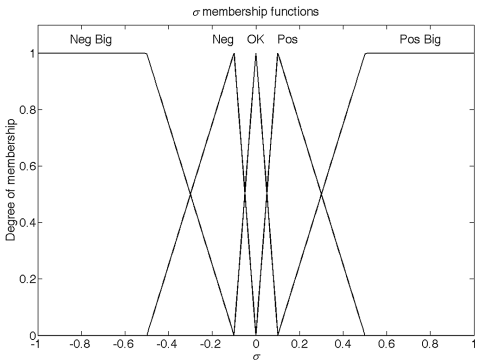

Fig. 6. Fuzzy membership functions for the σ angle.

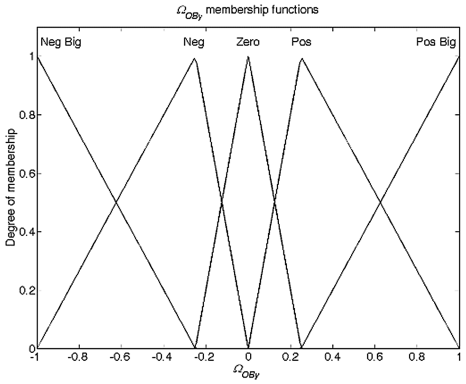

Fig. 7. Fuzzy membership functions for the orbital angular velocity $(\boldsymbol{\Omega_{OBy}})$ variable for the Y axis.

Nichols manual method [22]. This method usually gives good results as a first approximation in a large number of problems, however, the response obtained is considered too slow for the objectives of the mission, so an ad hoc method of calibration for this mission was devised.

The calibration method consisted in establishing all the gains, simulating a determined maneuver in each axis separately and analyzing the response to improve the gains in the next iteration, following a semiautomatic method. The maneuver chosen was a step of fifty degrees on the axis analyzed, leaving the rest in the nominal position, at the initial orbit of 400 km. The chosen angle is maintained to zero degrees during the first 100 s of the simulation and then, a step of 50° is commanded. The attitude and actuations commanded by the controller for the following 200 s were recorded. This maneuver was chosen to allow some maneuverability but also, to limit the simulation time.

From all the possible sets of gains, only those that comply with the following conditions for the error and its standard deviation were considered:

$$\bar{e} = \frac{\sum_{t=150s}^{t=200s} |e|}{N} \leq 0.01 \text{ rad} = 0.57\,^{\circ} \tag{7}$$

$$\sigma_e = \left[\left(\sum_{t=150s}^{t=200s} (e - \bar{e})^2 \right) / (N-1) \right]^{\frac{1}{2}} \leq 0.11\,^{\circ} \tag{8}$$

being the instantaneous error defined as:

$$e = 2\text{acos}(\boldsymbol{q_E}(4)) \tag{9}$$

Conditions (7) and (8) assure that the response is stable in time and with achievable accuracy. Then, the optimal gains set was the smallest cumulative cost one, which is the integral of the instantaneous cost, defined using the absolute values of all the actuations:

$$C = K_w |\Delta\omega_w| + K_T \sum |T_{cj}| \tag{10}$$

where K_w is the power needed for changing the wheel speed; which is simplified as the maximum power consumption over the maximum acceleration achievable by the wheel, and K_T is the inverse of a reference magnetic field value. For the hardware and mission studied, the values of those parameters are:

$$K_w \sim 4.9 \cdot 10^{-3} \text{ W/(rpm/s)}$$

$$K_T \sim 3.3 \cdot 10^{4} \text{ W/Nm}$$

The gains selected after the tuning process are reflected in Table 1. Note that this set of gains was obtained to have low power consumption.

Table 1. Gain values

K_γ	K_σ	K_μ	$K_{\Omega OBx}$	$K_{\Omega OBy}$	$K_{\Omega OBz}$	$K_{T_{cx}}$	K_ω	$K_{T_{cz}}$
34.38	6.86	401.07	26.18	13.79	113.45	4.2×10^{-6}	-0.44	1.89×10^{-5}

In the following sections we proceed with the optimization of the fuzzy control system gains by GA. First, in Sect. 5, we focus on the accuracy, so the function to be minimized is the error between the desired attitude and the actual attitude of the satellite. Second, in Sect. 6, we proceed to a multiobjective optimization of the gains where both, the accuracy and the power consumption are taken into account. So, depending on the operation modes, we obtain different set of gains that can be commanded to achieve the best performance of the controller and, of course, of the satellite, along the whole mission.

5 Gains Monoobjective Optimization

In this section, we proceed to the optimization of the gains of the fuzzy control system using GA and fulfilling, a single objective. To do this, it will be necessary to define an objective function to minimize, that introduces the variables that we are optimizing. Afterwards, values for the optimization method parameters have to be specified. Finally, after the optimization, results have to be analyzed and conclusions about their validities in a real system have to be drawn.

The first step for the objective function design is to define the maneuver that will be used to optimize. To facilitate comparisons between the optimized system and the system designed in [21], the same maneuver is used. This maneuver sets the desired angle at 0° during the firsts 100 s, then, commands a 50° step jump in the desired angle and simulates another 200 s, which means a total of 300 s of simulation. The objective function is defined as the sum of the instantaneous errors, defined in (9), in each instant along the maneuver. Table 2 summarizes the selection of parameters made for this monoobjective optimization.

Table 2. First monoobjective optimization parameters

Constraints	Bounds	[0 0 0; 1000 1000 1000]
Population	Population size	15
	Initial range	[0 0 0; 500 500 500]
Selection	Selection function	Tournament
	Tournament size	4
Reproduction	Elite count	2
	Crossover fraction	0.7
Mutation	Mutation function	Gaussian
Crossover	Crossover function	Scattered
Stopping criteria	Maximun number of generations	150
	Stall generations	30
	Function tolerance	10^{-9}

Both, the size of the population and the total number of generations, have a very large impact on the computational cost. To have a large number of generations and a small population size is better than the opposite. The firsts generations with small population size will, probably, give bad results, but the process of crossover and elite selection will make that, after a certain number of generations, almost the entire population will give a good result. For this reason, a population size of 15 and a limit of 150 generations have been chosen.

Concerning reproduction parameters, a crossover ratio of 0.7 has been chosen (based on references [9] and [23]) and an elite selection has been used, where the two best individuals from each generation are copied into the next generation.

Once the objective function and the optimization parameters have been defined, the optimization is performed obtaining the gains in Table 3.

Table 3. Gain values from monoobjective optimization

K_γ	K_σ	K_μ	$K_{\Omega OBx}$	$K_{\Omega OBy}$	$K_{\Omega OBz}$	$K_{T_{cx}}$	K_ω	$K_{T_{cz}}$
130.84	0.729	150.54	17.12	0.19	32.98	5.58	−196.93	193.35

Next, the details of the actuations per axis are described. Figures 8 and 9 show the results for roll angle. As it can be seen in Fig. 8, the attitude response of the satellite with the optimized gains has a nice performance, since it responds quickly, in less than 50 s, and it is placed in the desired position without stationary error.

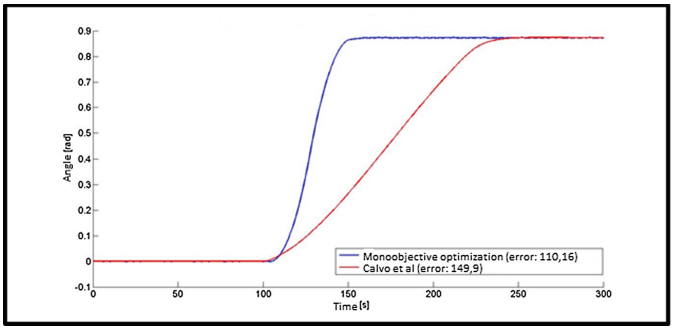

Fig. 8. Comparison of roll angle from the monoobjective optimization (blue line) and [21] (red line).

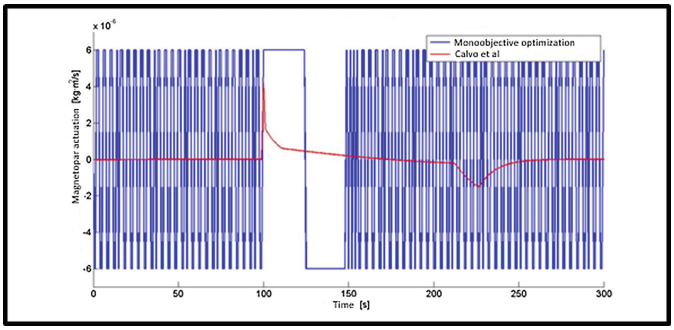

Fig. 9. Comparison of magnetorquer actuation for roll angle from the monoobjective optimization (blue line) and [21] (red line).

Table 4 summarizes the results of the optimization per axis and includes the results of [21] to compare. The comparison shows that the error is considerably smaller than the error in [21]. However, Fig. 9 shows that the control with the optimized gains is so abrupt that it is saturating the magnetorquer constantly. Therefore, this optimized control is too abrupt and it consumes a lot of electrical power, as Table 4 shows.

Table 4. Error and cost of monoobjective optimization and [21]

Method	Angle	Error (rad)	Cost (W)
Calvo et al. [21]	Roll	149.9	2.45
Monoobjective optimization	Roll	110.16	120.48
Calvo et al. [21]	Pitch	117.96	1.79
Monoobjective optimization	Pitch	94.76	136.9
Calvo et al. [21]	Yaw	150.58	6.41
Monoobjective optimization	Yaw	127.27	120.48

Figures 10 and 11 show the results for pitch angle. Figure 10 shows that the optimized control is a fast damped control. Although the over-impulse can be excessive, the control seems to be much better than in [21] in terms of accuracy.

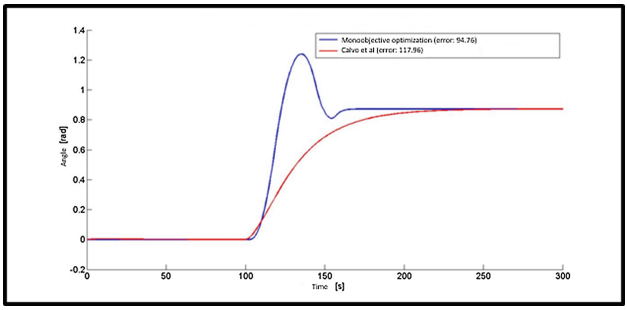

Fig. 10. Comparison of rpitch angle from the optimization (blue line) and [21] (red line).

Concerning the acceleration of the moment wheel, Fig. 11 shows that, when the step occurs, it saturates at maximum deceleration and then saturates at maximum acceleration to control the angular velocity of the satellite. So, once again, this optimized control is too abrupt and consume a high electric power, although it is not as evident as in the case of the roll axis.

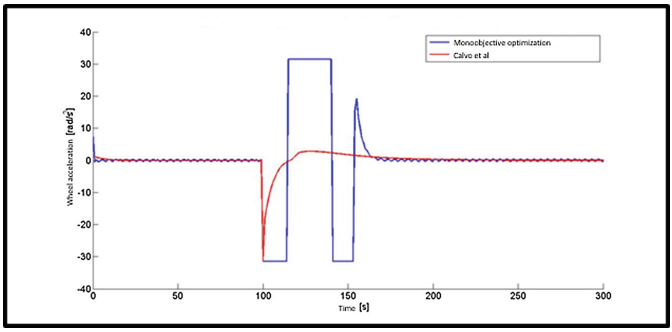

Fig. 11. Comparison of the wheel acceleration from the optimization (blue line) and [21] (red line).

Results for yaw angle are not presented as they are very similar to roll angle.

To conclude, the optimized control is too abrupt and involves a consumption of electrical power that does not justify the improvement in accuracy. This means that, to have a truly optimized controller, the objective function must include in some way the cost of electrical power, in addition to the error function.

6 Gains Multiobjective Optimization

The results obtained in the previous section show that the optimization of the gains of the control system by fuzzy logic, with the only objective of error minimization, gives an inadmissible action: although the error is small, it seems unlikely that the real satellite will be able to maintain this behavior in its actuators constantly, without adding the tremendous cost of electric power.

Therefore, it is necessary to perform a multiobjective optimization, in which, in addition to minimizing the error, the cost of electric power is minimized. To do this, the objective function of the previous section must be redesigned. In this case, the outputs of this function are the error, as calculated in Sect. 5, and the cost, calculated as the integral of the instantaneous cost defined in (10). Pareto frontier will be estimated to analyze the most interesting points for the satellite performance. From all the possible sets of gains, only those that comply with conditions in (7) and (8) for the error and its standard deviation are considered.

Table 5 summarizes the selection of parameters made for three different optimizations. By performing the three optimizations of Table 5, three Pareto fronts are obtained, the gains obtained are presented in Table 6 and the main results of the optimizations are summarized in Table 7. Next, results per each axis are commented.

Table 5. Multioobjective optimizations parameters

	Optimization	First	Second	Third
Population	Population size	30	30	60
Selection	Selection function	Tournament	Tournament	Tournament
	Tournament size	2	2	2
Reproduction	Crossover fraction	0.7	0.8	0.8
Mutation	Mutation function	Gaussian	Constraint dependent	Constraint dependent
Crossover	Crossover function	Scattered	Intermediate	Intermediate
Multiobjetive setting	Pareto front fraction	0.7	0.7	0.5
Stopping criteria	Maximun number of generations	300	300	400
	Stall generations	60	60	60
	Function tolerance	10^{-6}	10^{-6}	10^{-6}

Table 6. Gain values from multiobjective optimizations

Roll angle			
	K_γ	$K_{\Omega OBx}$	$K_{T_{cx}}$
Point 1	716.05	80.89	1.77×10^{-5}
Point 2	599.69	61.83	2.32×10^{-5}
Point 3	588.37	40.93	3.52×10^{-5}
Pitch angle			
	K_σ	$K_{\Omega OBy}$	K_ω
Point 1	0.85	1.68	−1.90
Point 2	0.47	1.20	−1.82
Point 3	8.33	11.44	−0.54
Yaw angle			
	K_μ	$K_{\Omega OBz}$	$K_{T_{cz}}$
Point 1	0.74	0.56	1.27×10^{-3}
Point 2	0.61	0.67	2.07×10^{-4}
Point 3	0.80	4.29	0.01

Table 7. Error and cost from multiobjective optimizations and from [21]

Method	Angle	Error (rad)	Cost (W)
Calvo et al. [21]	Roll	149.9	2.45
Optimization point 1	Roll	132.53	2.92
Optimization point 2	Roll	124.2	3.97
Optimization point 3	Roll	116.1	6.64
Calvo et al. [21]	Pitch	117.96	1.79
Optimization point 1	Pitch	117.26	1.7
Optimization point 2	Pitch	124.7	1.32
Optimization point 3	Pitch	110.0	6.09
Calvo et al. [21]	Yaw	150.58	6.41
Optimization point 1	Yaw	135.4	11.03
Optimization point 2	Yaw	131.9	15.09
Optimization point 3	Yaw	126.3	18.15

6.1 Roll Results

Figure 12 shows the Pareto frontier for the three optimizations, together with the point corresponding to [21]. As can be seen, the results of [21] are really good, and none point which improves it in both objectives has been found. However, by slightly increasing the cost of the maneuver, much lower error values are obtained as, for example, Point 1 in Table 7.

To analyze the behavior of the optimized controllers, three points have been chosen: a point of low consumption (Point 1 in Tables 6 and 7), an intermediate point (Point 2) and a point with a low error, but whose power consumption is reasonable (Point 3).

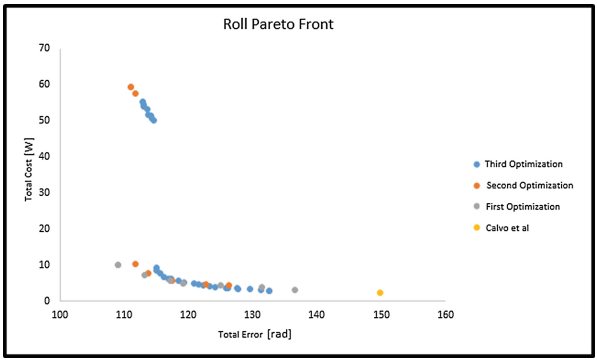

Fig. 12. Pareto frontier for roll angle.

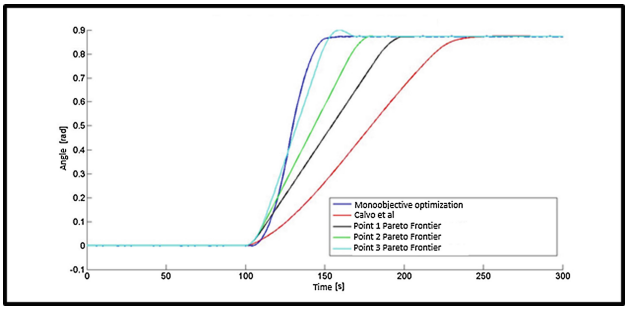

Fig. 13. Comparison of roll angle for different operation points and [21].

Figure 13 shows that the solutions from Pareto frontier represent an intermediate attitude between the fastest possible, which is optimized only with the aim of minimizing the error, and [21]. On the other hand, Figs. 14 and 15 shows that the cost of electrical power is similar to that of [21] and totally different from that of the

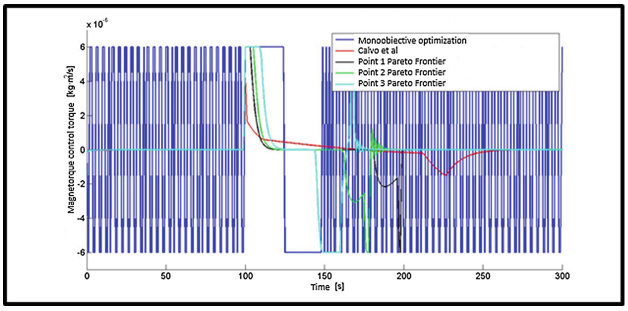

Fig. 14. Comparison of magnetorquer actuation for roll angle for different operation points and [21].

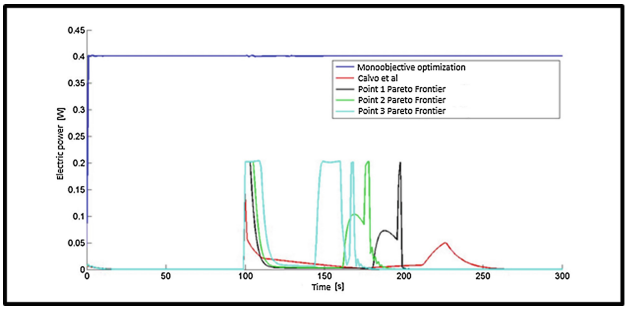

Fig. 15. Comparison of cost for roll angle for different operation points and [21].

monoobjective optimization. The behavior of the selected points is not based on saturating the magnetorquer from one extreme to the other, giving a much more realistic control attitude performance.

6.2 Pitch Results

Figure 16 shows the Pareto frontier for the three optimizations, together with the point corresponding to [21]. In this case, even though the manual adjustment is very good, a point has been found that improves it in both objectives: Point 1 in Table 7.

To analyze the behavior of the points resulting from the optimization, three of them are selected: a point that improves in both objectives to [21] (Point 1 in Tables 6 and 7), a point of low consumption (Point 2) and a point with low error value, but with a moderate power consumption (Point 3).

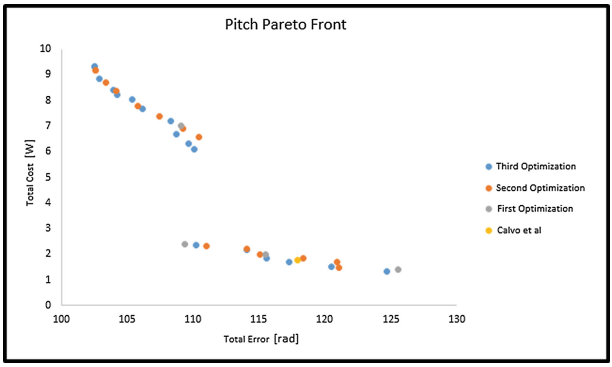

Fig. 16. Pareto frontier for pitch angle.

Figures 17, 18 and 19 shows the results of the optimizations. Figure 17 shows that the behavior of the Pareto frontier selected points and [21] are very similar. Concerning the actuations, for Points 1, 2 and [21] it would not be necessary to use the magnetorquer, as the wheel is never saturated (see Fig. 18). However, the monoobjective

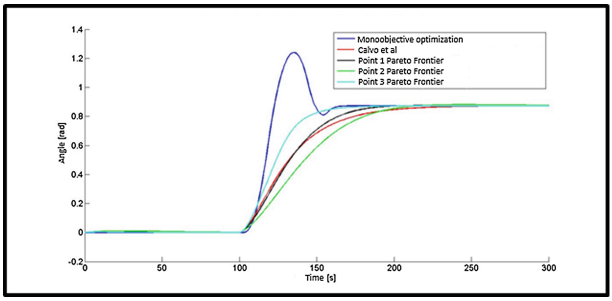

Fig. 17. Comparison of pitch angle for different operation points and [21].

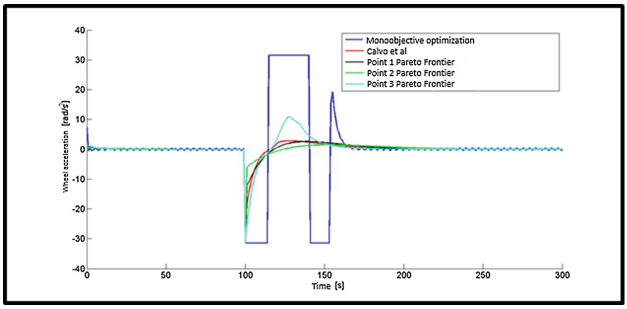

Fig. 18. Comparison of wheel acceleration for different operation points and [21].

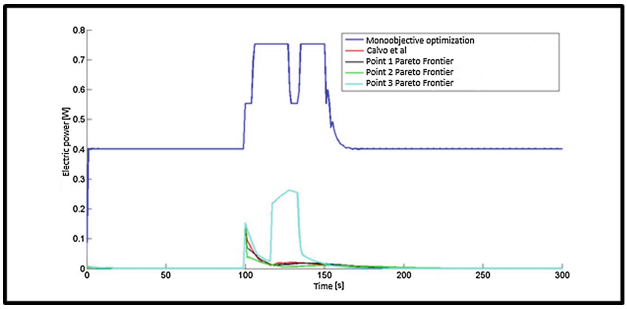

Fig. 19. Comparison of cost for pitch angle for different operation points and [21].

optimization provides such aggressive control, that it becomes necessary to activate the magnetorquer. The same happens with Point 3. This is the main cause of the differences of the cost function in Fig. 19.

6.3 Yaw Results

Finally, the same three optimizations are carried out to the yaw axis. In this case no good results in terms of cost have been found. Figure 20 shows the Pareto frontiers.

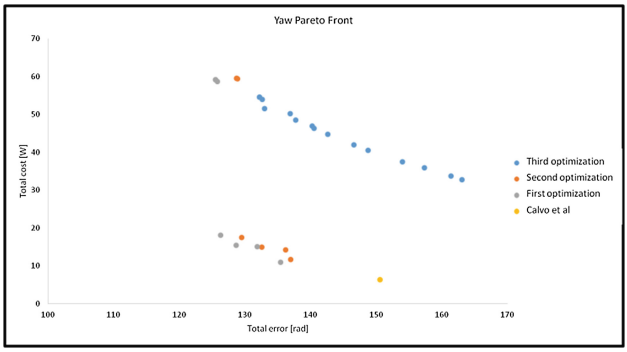

Fig. 20. Pareto frontier for yaw angle.

To analyze the optimization behavior, three points have been chosen that represent a point of low consumption (Point 1), an intermediate point (Point 2) and a point with a low error, but whose power consumption is reasonable (Point 3).

Figures 21, 22 and 23 shows the results of the optimizations. Figure 21 shows that the response of the multiobjective optimization points is considerably faster than that

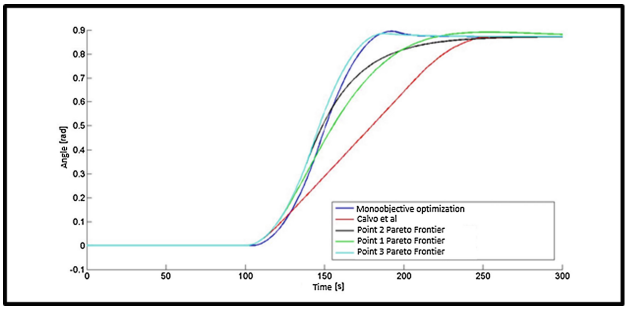

Fig. 21. Comparison of yaw angle for different operation points and [21].

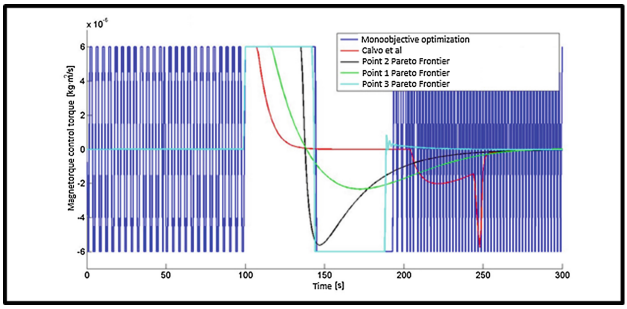

Fig. 22. Comparison of magnetorquer actuation for yaw angle for different operation points and [21].

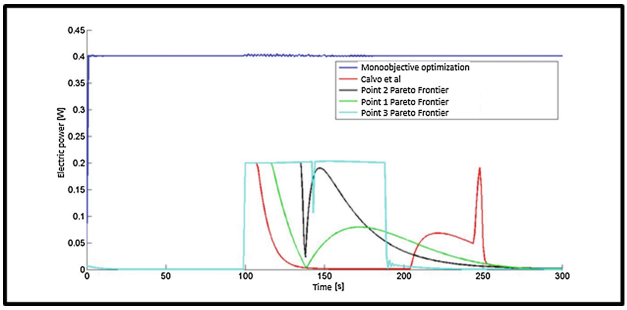

Fig. 23. Comparison of cost for yaw angle for different operation points and [21].

obtained from [21]. This is achieved by saturating the magnetorquer during a greater time interval (see Fig. 22), increasing the electrical power cost. Notice also that Point 3 achieves a lower error than monoobjective optimization with much lower power consumption.

7 Conclusions

Summing up, this study shows that the optimization of the fuzzy logic control system gains must be considered as a multiobjective problem, since the monoobjective optimization leads to results that, although they minimized the error, they are not applicable in real systems due to its high cost of electrical power and the difficulty of applying such abrupt control in flight.

For roll axis, interesting sets of gains have been found, because with a slight increase in electric power with respect to [21], they can significantly reduce the error between the desired position and the real one during the design maneuver. For the pitch axis, a point has been found that improves in both objectives, the one obtained in [21] and that has a quite similar behavior. Finally, in the yaw axis, the optimizations have not been good enough and, in general terms, we have found points that reduce the error considerably, but also increase considerably the cost with respect to the control in [21].

Therefore, this study shows that the control solution in [21] is a good one in terms of cost. In addition, with GA multiobjective optimizations, several points with an optimal combination of costs and errors were obtained, which allows the flexibility of changing the controller gains to fasters or lower costs ones depending on the necessity of the real system. Then, it would be possible to save multiple sets of gains in the on-board computer memory to generate a new algorithm which, depending on the batteries state of charge or the maneuver requirement, could select certain values of the gains in order to optimize the attitude control performance. This situational evaluation option has a tiny computational cost for the algorithm and, also, to keep multiple values of gains would be almost irrelevant in the computer memory cost.

Finally, regarding the future work related to the research presented, the following is worth to be mentioned: (i) Not only the gains but also the membership functions can be redefined using genetic algorithms, maintaining the general rules (e.g. symmetry) to

obtain even better fuzzy solutions if possible. (ii) The PID gains could also be redefined using genetic algorithms to compare again both controllers.

References

1. Harland, D.M., Lorenz, D.R.D.: Space Systems Failures. Praxis (2005)
2. Wingrove, R.C.: A Study of Guidance to Reference Trajectories for Lifting Reentry at Supercircular Velocity. National Aeronautics and Space Administration, Whashington (1963)
3. Sidi, M.J.: Spacecraft Dynamics and Control: A practical Engineering Approach. Cambridge University Press (2000)
4. Gadelha de Souza, L.C.: Design of satellite control system using optimal nonlinear theory. Mech. Based Des. Struc. Mach. **34**(4), 351–364 (2006)
5. Ortega, G.: Fuzzy logic techniques for rendezvous and docking of two geostationary satellites. Telematics Inform. **12**(3–4), 213–227 (1995)
6. Steyn, W.H.: Comparison of low-earth-orbit satellite attitude controllers submitted to controllability constraints. J. Guid. Control Dyn. **17**(4), 795–804 (1994)
7. Nagi, F., Ahmed, S., Abidin, A., Nordin, F.: Fuzzy bang-bang relay controller for satellite attitude control system. Fuzzy Sets Syst. **161**, 2104–2125 (2010)
8. Guana, P., Liub, X.-J., Liub, J.-Z.: Adaptive fuzzy sliding mode control for flexible satellite. Eng. Appl. Artif. Intel. **18**, 451–459 (2005)
9. Walker, A.R., Putman, P.T., Cohen, K.: Solely magnetic genetic/fuzzy-attitude-control algorithm for a cubeSat. J. Spacecr. Rockets **52**(6), 1627–1639 (2015)
10. Cheng, C., Shu, S., Cheng, P.: Attitude control of a satellite using fuzzy controllers. Expert Syst. Appl. **36**, 6613–6620 (2009)
11. Zou, A., Dev Kumar, K., Hou, Z.: Quaternion-based adaptive output feedback attitude control of spacecraft using Chebyshev neural networks. IEEE Trans. Neural Netw. **21**(9), 1457–1471 (2010)
12. Fazlyab, A.R., Saberi, F.F., Kabganian, M.: Adaptive attitude controller for a satellite based on neural network in the presence of unknown external disturbances and actuator faults. Adv. Space Res. **57**(1), 367–377 (2016)
13. Zadeh, L.A.: Fuzzy sets. Inform. Control **8**, 338–353 (1965)
14. Lee, C.C.: Fuzzy logic in control systems: Fuzzy logic controller-part I. IEEE Trans. Syst. Man Cybern. **20**, 404–418 (1990)
15. Jantzen, J.: Foundations of Fuzzy Control: A Practical Approach, 2nd edn. Wiley (2013)
16. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. Syst. Man, Cybern. **1**, 116–132 (1985)
17. Østergaard, J.: Fuzzy logic control of heat exchanger process. In: Gupta, M.M., Saridis, G. N., Gaines, B.R. (eds.) Fuzzy Automata and Decision Processes, pp. 285–320. North-Holland, Amsterdam (1977)
18. Oshima, H., Yasunobu, S., Sekino, S.I.: Automatic train operation system based on predictive fuzzy control. IEEE Int. Workshop Artif. Intell. Ind. Appl., 485–489 (1988)
19. Toshikazu, T., Toshiharu, H.: A practical application of fuzzy control for an air-conditioning system. Int. J. Approx. Reason. **5**, 331–348 (1991)
20. Layne, J., Passino, K.: Fuzzy model reference learning control for cargo ship steering. IEEE Control Syst. **13**(6), 23–34 (1993)

21. Calvo, D., Avilés, T., Lapuerta, V., Laverón-Simavilla, A.: Fuzzy attitude control for a nano satellite in low earth orbit. Expert Syst. Appl. **58**, 102–118 (2016)
22. Ziegler, J.G., Nichols, N.B.: Optimum settings for automatic controllers. J. Dyn. Syst.-T. ASME **115**(2B), 759–765 (1993)
23. Hou, L.R., Yi, Z.: Fuzzy logic controller based on genetic algorithms. Fuzzy Sets Syst. **83**, 1–10 (1996)