




# Optimization of Type-2 Fuzzy Logic Controller Design Using the GSO and FA Algorithms

Emer Bernal<sup>1</sup> · Marylu L. Lagunes<sup>1</sup> · Oscar Castillo<sup>1</sup>  · José Soria<sup>1</sup> · Fevrier Valdez<sup>1</sup>

Received: 7 January 2020 / Revised: 17 September 2020 / Accepted: 28 September 2020 / Published online: 21 November 2020  
© Taiwan Fuzzy Systems Association 2020

**Abstract** This paper presents a comparative study between the firefly algorithm (FA) and the galactic swarm optimization (GSO) method, where the performance of both methods is observed and tested in the optimization of a fuzzy controller for path tracking of an autonomous mobile robot. The main contribution of this work is finding the best method that generates an optimal vector of values for the membership function optimization of the fuzzy controller. This with the goal of improving the performance of the controller and thus the trajectory generated by the autonomous robot is closer to the desired trajectory. It should be noted that the fuzzy controller that is optimized is an interval type-2 fuzzy controller, which has a greater capability for managing uncertainty than a type-1 fuzzy controller. In this case, the limiting membership functions in the interval type-2 fuzzy sets are themselves type-1 fuzzy sets that define the footprint of uncertainty. Type-2 fuzzy controllers have been shown in previous works to handle better the control of robotic systems under noisy and dynamic conditions and this is why their optimal design is very important. Simulation results show that GSO outperforms FA in the optimal design of interval type-2 fuzzy controllers.

**Keywords** Firefly algorithm · Galactic swarm optimization · Fuzzy controller · Type-2 fuzzy systems · Fuzzy sets

## 1 Introduction

Nowadays metaheuristics are being used for optimization of fuzzy controllers, and these controllers need to be optimized because in many cases they do not achieve the optimal performance that is required for real-world applications. In most cases, these controllers are called type-1 fuzzy logic controllers (FLC) because they use the original definition of fuzzy sets [1]. Fuzzy logic was originally proposed by Lotfi Zadeh in [2], where the concepts of fuzzy sets and fuzzy logic were introduced. In this case, the elements of a set are given a numeric value as a measure of the uncertainty in the so-called membership functions. The membership functions are defined for a linguistic variable of a fuzzy set. In addition to the original fuzzy logic (type-1) that was proposed from the beginning, type-2 fuzzy logic was also later developed, and arises with the purpose of solving more complex problems, that is, with a higher level of uncertainty, when compared to the problems that can be solved with type-1 fuzzy logic [3].

Type-2 fuzzy sets were also introduced by Zadeh as an extension of ordinary fuzzy sets (type-1). In a type-2 fuzzy set the membership degrees are also fuzzy. In this sense, a type-1 fuzzy set is a special case of a type-2 fuzzy set because its secondary membership function is a subset with a single element [4].

Type-2 fuzzy logic systems can be viewed as a collection of type-1 fuzzy logic systems; therefore, their ability for uncertainty management is greater. The works in [5–8] describe the use of type-2 fuzzy systems that provide very good results in different control problems. In [9] the theoretical advances of type-2 fuzzy controllers are described, and also what remains to be done in this area was outlined. On the other hand, in [10] the difference between a type-1 and a type-2 fuzzy controller is explained in a very detailed

---

✉ Oscar Castillo  
ocastillo@tectijuana.mx

<sup>1</sup> Tijuana Institute of Technology, Tijuana, BC, Mexico

fashion. In addition, this work described some limitations of type-1 fuzzy controllers to solve problems with higher levels of uncertainty, and the applications and benefits of type-2 fuzzy controllers are also outlined.

Both type-1 and type-2 fuzzy systems have been previously optimized with metaheuristics, for example, the optimization of type-1 fuzzy controllers is found in [11–13], which deal with the optimization of fuzzy controllers of autonomous mobile robots using the firefly algorithm. Also in [14], GSO was used for the optimization of a fuzzy controller of an autonomous robot following a trajectory, where the dynamic adjustment of the most important parameters for the operation of the GSO algorithm is presented. In addition, in [15] the GSO algorithm was used in the optimization of the water tank fuzzy controller.

There are many fuzzy controller optimization papers, using other metaheuristics. For example, in [16] the genetic algorithm (GA) is used to evolve the architecture of a type-2 fuzzy controller in robot navigation for real environments. Other authors have used fuzzy controllers of autonomous robots following a trajectory as described in [17], also there are more fuzzy controller applications thanks to their efficiency and performance, as illustrated in [18, 19], where a comparison is made between type-1 and type-2 fuzzy controllers, and in [20] two fuzzy controllers are used to control the liquid-level process in a tank.

The main motivation of this work is to introduce, through the use of a metaheuristic algorithm, an optimization method, where optimal performance can be obtained for achieving satisfactory results in the control of an autonomous mobile robot. It has been previously observed that the use of parameter adjustment in metaheuristic algorithms for the optimization of mathematical functions and control problems provides competitive results, so in this work we propose using fuzzy logic to perform the dynamic adjustment of the parameters of the algorithms and measure their performance applied in the optimization of the fuzzy controller of the autonomous mobile robot.

The main contribution is the proposed optimization methodology, as well as finding the optimal parameters for the optimization of a type-2 fuzzy controller. In this case, the control of the behavior of a robot following a trajectory is considered, which aims at following a given trajectory with a margin of error as low as possible, calculated by means of an established metric. In addition, a comparison of the optimization efficiency between two metaheuristic algorithms, the firefly algorithm (FA) and galactic swarm optimization (GSO) is considered. The reason for considering FA and GSO is that both algorithms have previously shown good performance when applied to fuzzy controller optimization problems.

The organization of this paper is as follows: Sect. 2 details the inspiration and equations required by the GSO and FA algorithms to perform their search and optimization. In Sect. 3 the proposed approach for the optimization of the type-2 fuzzy controller of the autonomous mobile robot is presented. In Sect. 4 the case study of the autonomous mobile robot is presented. Section 5 shows the results obtained from the experiments developed with the proposed approach for the autonomous mobile robot plant. In Sect. 6, a discussion about the obtained results is presented. Finally, in Sect. 7 conclusions and future work are outlined.

## 2 Background

In a review of the state of the art, some articles focused on optimizing membership functions with FA and GSO were found, but none of these works use the same parameters of experimentation that are proposed here. Since the combination of these metaheuristics is rare in the literature, we will compare the performance results of these algorithms for this type of optimization problems.

In this section, we are presenting the relevant theory and concepts for this work.

### 2.1 Firefly Algorithm (FA)

This method was inspired by the flickering fireflies, where each firefly is attracted to another depending on the intensity of light or brightness it generates. It was developed in 2008 by Xin She Yang [21, 22]. The firefly algorithm has three main rules: Rule 1 stipulates that any firefly can be attracted to another firefly regardless of their sex, since in this method the fireflies are unisex. Rule 2 describes that the less glowing fireflies will be attracted to the brightest firefly. Finally, Rule 3 states that the search space is described by the objective function.

The FA has three equations that represent the movement of the fireflies. The first one is called the attraction equation:

$$\beta = \beta_0 e^{-\gamma r^2}, \quad (1)$$

where  $\beta$  represents the attractiveness of a firefly;  $\beta_0$  is the initial attractiveness in  $r = 0 \in [0,1]$ ;  $e \approx 2.71828$  is the basis of the natural logarithms;  $r$  is the distance between each of two fireflies, and  $\gamma$  determines the variation of attractiveness as the distance increases between the fireflies  $\in [0, 1]$ .

The movement equation is as follows:

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2} (x_j^t - x_i^t) + \alpha_r \epsilon_i^t, \quad (2)$$

where  $x_i^{t+1}$  represents the next position;  $x_i^t$  is the actual position;  $\beta_0 e^{-\gamma r_{ij}^2} (x_j^t - x_i^t)$  represents the attraction;  $\alpha_t$  is the randomization with  $\alpha$  being the parameter of randomness  $\alpha \in [0, 1]$ , and  $\epsilon_i^t$  represents the vector of random numbers extracted from a Gaussian distribution.

The distance equation is expressed as:

$$r_{ij} = \sqrt{\sum_{k=1}^d (x_{i,k} - x_{j,k})^2}, \quad (3)$$

where  $r_{ij}$  represents the Euclidian distance between two fireflies  $i$  and  $j$ ;  $x_{i,k}$  is the  $k$ th component of the spatial coordinate;  $x_i$  is the  $i$ th firefly, and  $d$  is the number of dimensions.

## 2.2 Galactic Swarm Optimization (GSO)

Galactic swarm optimization (GSO) is a recently created algorithm, which was proposed in 2016 by Muthiah-Nakarajan and Noel [23]. Galactic swarm optimization uses multiple exploration and exploitation cycles to carry out its search process, thus offering greater opportunities to find the global minimum.

In galactic swarm optimization, the concept of a swarm is used, which is represented by a set of elements called particles, as in the PSO algorithm, since it was taken as inspiration to demonstrate this technique. In the GSO algorithm the population is divided into  $M$  subpopulations called subswarms. This metaheuristic consists of two levels [20]. In the first level, the method starts to explore independently in each of the subswarms, thus improving the search for possible solutions. Since the GSO algorithm uses the PSO algorithm as a basis, the velocity and position of the particles are calculated according to the number of iterations determined for the first level [24].

In the second level, a super swarm is formed based on the best individuals of each of the subswarms, so that in the end the best possible solution is obtained. In the same way the calculation of the velocity and position of the particle is used as in the level 1. The equations needed to calculate the velocity and position of the GSO algorithm particles are represented below [25].

Level 1:

$$V_j^{(i)} \leftarrow W_1 V_j^{(i)} + C_1 r_1 (P_j^{(i)} - X_j^{(i)}) + C_2 r_2 (g^{(i)} - X_j^{(i)}), \quad (4)$$

$$X_j^{(i)} \leftarrow X_j^{(i)} + V_j^{(i)}, \quad (5)$$

where  $V_j^{(i)}$  is the velocity of the individual;  $W_1$  represents the inertia;  $C_1$  and  $C_2$  are the acceleration constants;  $r_1$  and  $r_2$  are random numbers;  $P_j^{(i)}$  is the best personal associated

with the individual  $X_j^{(i)}$ ;  $g^{(i)}$  represents the best global of the subswarm  $X_i$ , and  $X_j^{(i)}$  position of the individual.

$$w_1 = 1 - \frac{k}{I_1 + 1}, \quad (6)$$

where  $k$  is the current iteration number, an integer ranging from 0 to  $I_1$ ; and  $I_1$  is the number of iterations of the first level.

Level 2:

$$Y^{(i)} \in Y : i = 1, 2, \dots, M \quad (7)$$

$$Y^{(i)} = g^{(i)},$$

$$V^{(i)} \leftarrow W_2 V^{(i)} + C_3 r_3 (P^{(i)} - Y^{(i)}) + C_4 r_4 (g - Y^{(i)}), \quad (8)$$

$$Y^{(i)} \leftarrow Y^{(i)} + V^{(i)}, \quad (9)$$

where  $V^{(i)}$  is the velocity of the individual;  $W_2$  represents the inertia in level 2;  $C_3$  and  $C_4$  are the acceleration constants in level 2;  $r_3$  and  $r_4$  are random numbers;  $P^{(i)}$  represents the personal best;  $g$  represents the global best, and  $Y(i)$  the position of the individual.

## 3 Proposal

The main contribution of this work is the proposed optimization methodology, as well as its application for finding the optimal parameters in the design of a type-2 fuzzy controller, which must control the behavior of an autonomous mobile robot. The fuzzy controller aims at following a given trajectory with a margin of error as minimum as possible, calculated by means of an established metric of control performance.

The most important difference between using the dynamic adjustment of parameters in metaheuristic algorithms with respect to the use of fixed parameters is that the parameters selected for dynamic adjustment are modified as the iterations proceed, which leads us to obtain better solutions.

The methodology consists on using a metaheuristic algorithm to perform the optimization of the membership functions parameters of the type-2 fuzzy robot controller for path following. In this case, we measure the type-2 fuzzy controller's performance and return a result according to the established performance metric and continue with the optimization until a stopping criterion or a previously established number of iterations are met. This methodology can be found in a summarized form in Fig. 1.

The optimization of the fuzzy controller is performed with the original algorithms and their variants using dynamic parameter adjustment in the GSO and FA

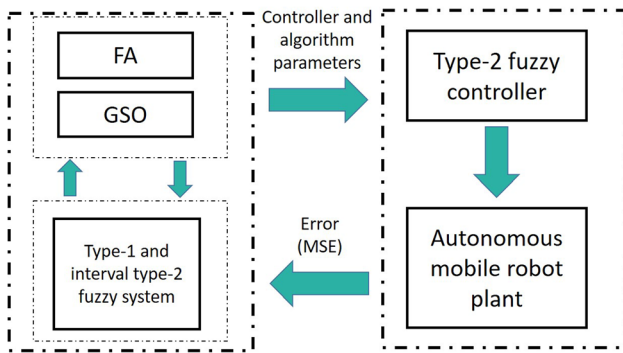


Fig. 1 Proposed optimization method

algorithms. In the first case, for the optimization variants we use type-1 fuzzy systems. The fuzzy system used for parameter adjustment of the GSO algorithm uses “Iteration” as input variable and as output variables the  $c_3$  and  $c_4$  parameters. Each of the variables is composed of three triangular membership functions labeled as “low”, “medium” and “high”, as can be seen in Fig. 2. The fuzzy system FA\_T1 that performs parameter adjustment of the FA uses the “Iteration” variable as input variable and as an output variable the  $\alpha$  parameter, and these variables are

composed of three triangular membership functions labeled as “low”, “medium” and “high”, as illustrated in Fig. 3.

In the second variant, interval type-2 fuzzy systems are used in FGSO IT2 and FA\_T2, which are an extension of the previously used type-1 fuzzy systems. The type-2 fuzzy systems are formed with the same input and output variables, which are partitioned in the same way and sharing the same form for the set of fuzzy rules. This fuzzy system can be found in Figs. 4 and 5 as shown, respectively.

Fuzzy system rules were designed with the idea that when the algorithms are in the initial iterations they can explore and when they are in the last iterations they can exploit.

Fuzzy rules for FGSO:

1. If (iteration is low) then ( $c_3$  is low) and ( $c_4$  is High),
2. If (iteration is medium) then ( $c_3$  is medium) and ( $c_4$  is medium),
3. If (iteration is high) then ( $c_3$  is high) and ( $c_4$  is low).

Fuzzy rules for FA:

1. If (iteration is low) then (alpha is low),
2. If (iteration is medium) then (alpha is medium),
3. If (iteration is high) then (alpha is high).

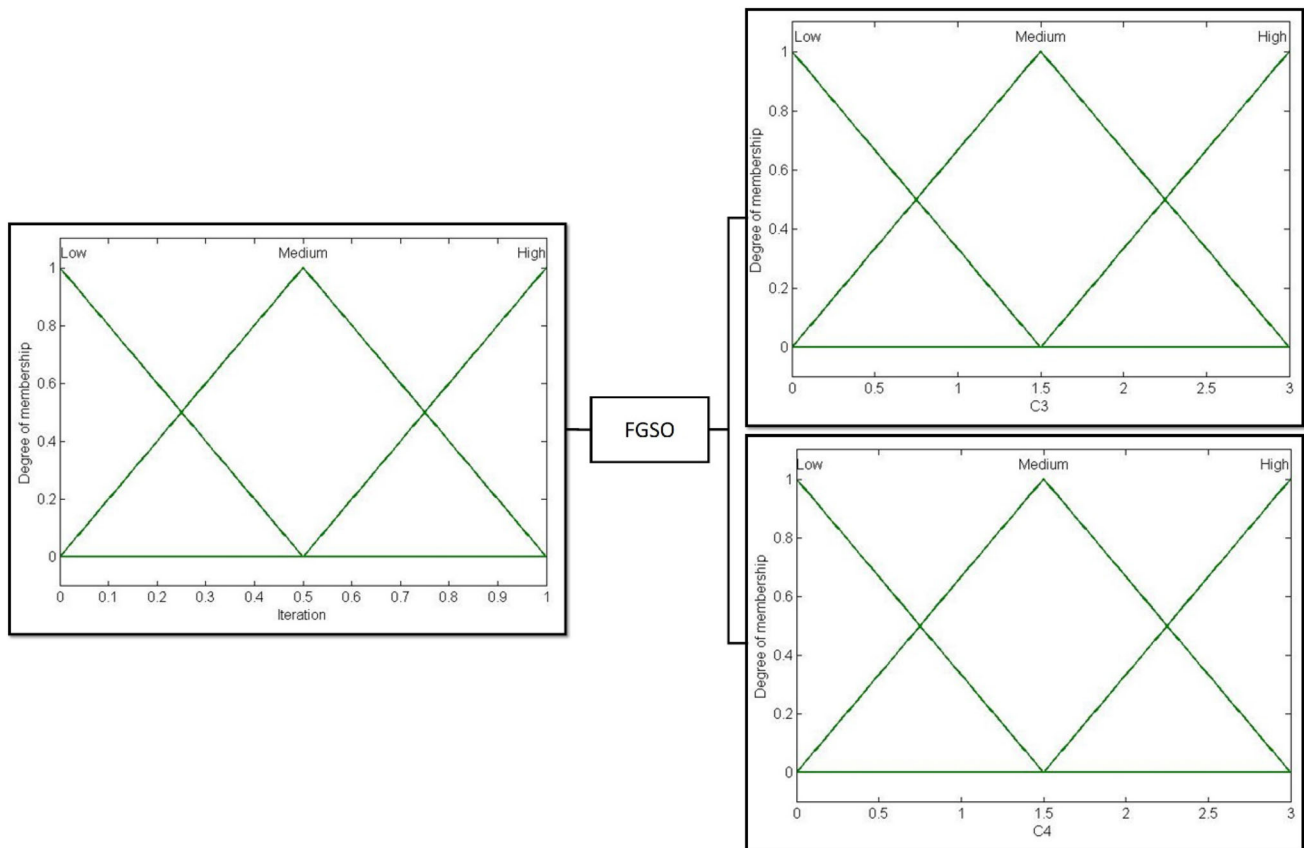


Fig. 2 Type-1 fuzzy system for the GSO algorithm

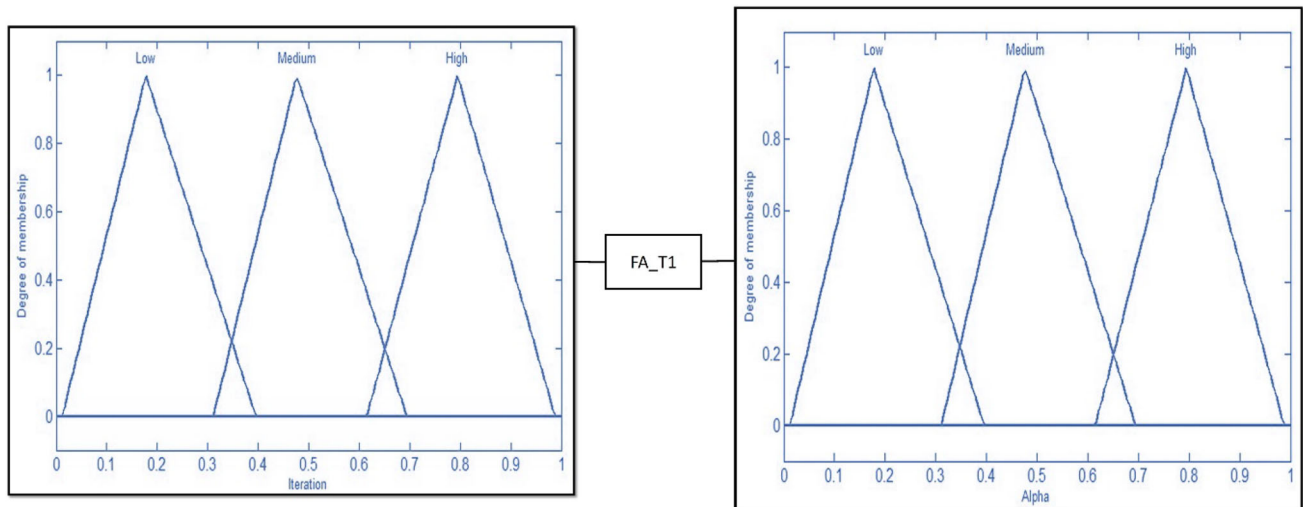


Fig. 3 Type-1 fuzzy system for the FA

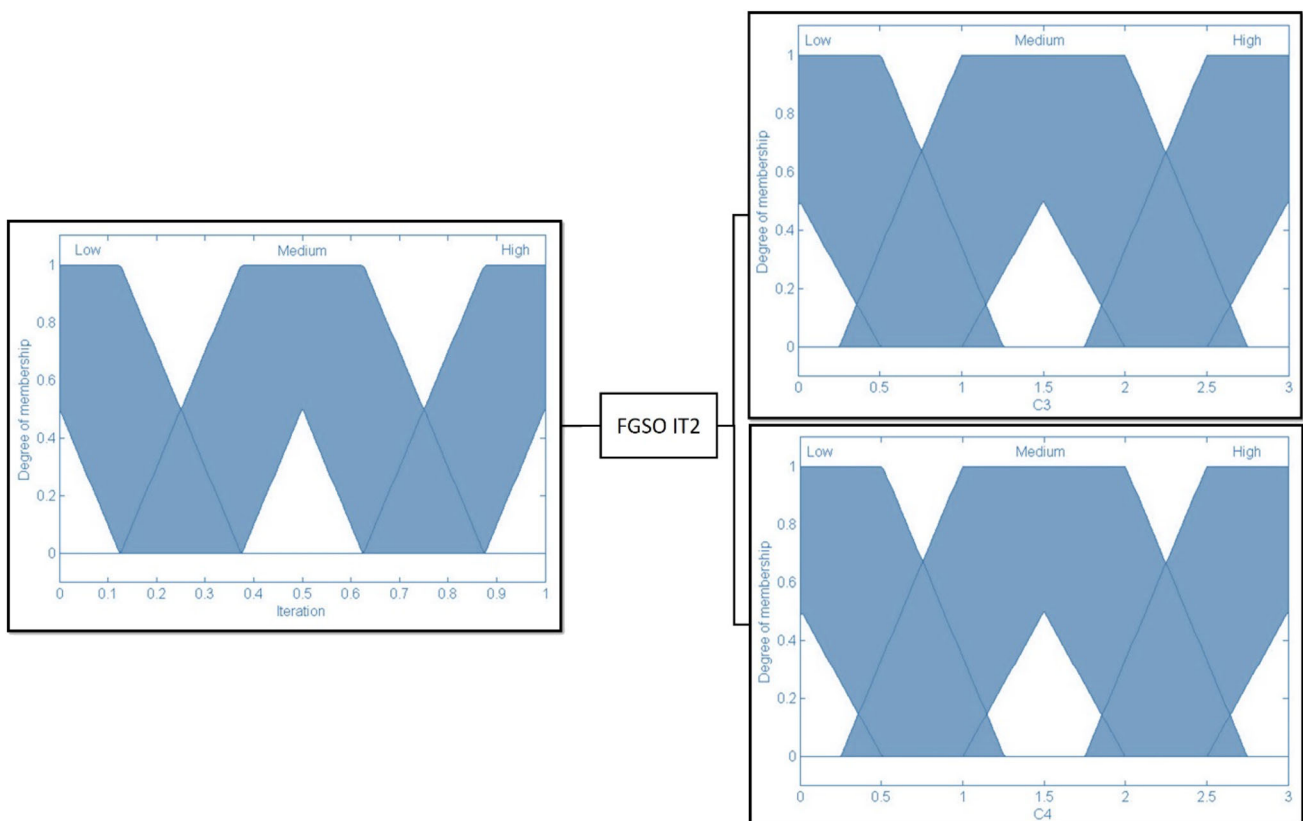


Fig. 4 Interval type-2 fuzzy system for the GSO algorithm

Each individual or possible solution that is used in our proposal is formed by the required parameters to automatically generate the type-2 fuzzy controller avoiding manually generating the fuzzy controller, in this way expecting to obtain better results than those obtained using the basis fuzzy controller. In Fig. 6, we can find the way in

which individuals are composed in the metaheuristic algorithms [26–28].

Each individual contains the necessary parameters to form the fuzzy controller. In this case, the controller has two input variables, which are formed by two trapezoidal membership functions and this correspond to 8 parameters to form each and a triangular with six parameters giving a

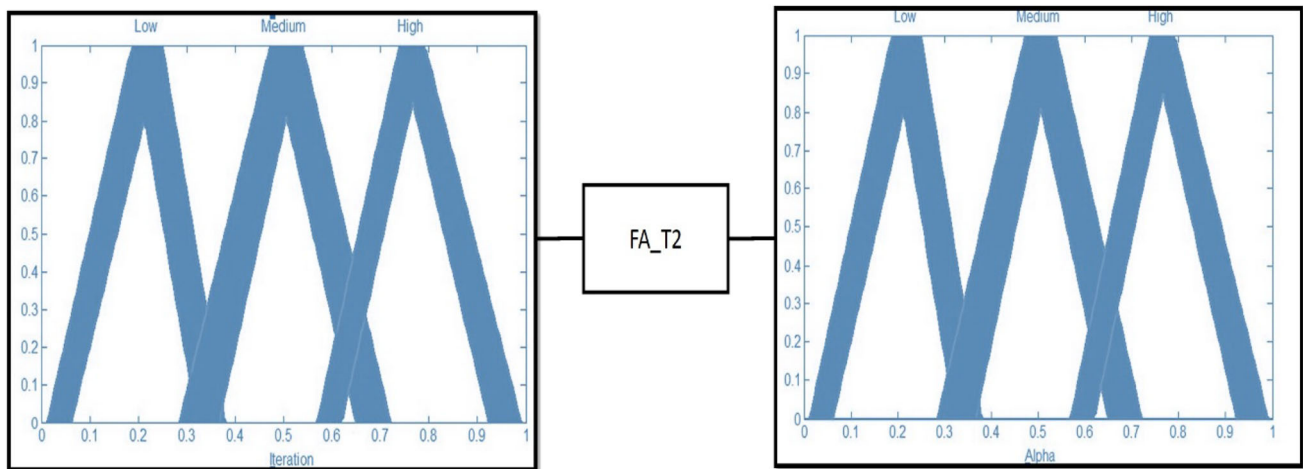


Fig. 5 Interval type-2 fuzzy system for the FA

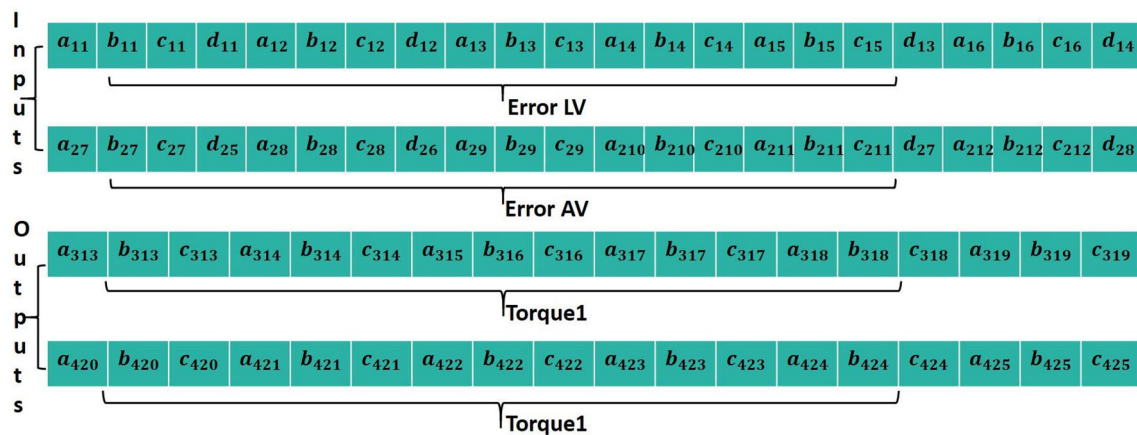


Fig. 6 Structure of individuals in the GSO algorithm and FA

total of 22 parameters per input variable. The output variables consist of three triangular membership functions which are formed with six parameters each, giving a total of 18 parameters per output variable. Once all the parameters are obtained to form the input and output variables, the fuzzy controller is built.

The structure of the individuals in the GSO and FA algorithms help us to form the interval type-2 fuzzy controllers, which are formed by triangular and trapezoidal membership functions, and their equations and graphical representation using type-2 fuzzy logic are presented below.

The representation of the interval type-2 triangular membership functions (itritype2) of the fuzzy controller consists of 6 parameters  $a_{n1}$ ,  $b_{n1}$ ,  $c_{n1}$ ,  $a_{n2}$ ,  $b_{n2}$  and  $c_{n2}$ , where the parameter values satisfy  $a_{n1} < a_{n2}$ ,  $b_{n1} < b_{n2}$  and  $c_{n1} < c_{n2}$  and this is shown as follows:

$$\begin{aligned} \mu(x) &= \text{itritype2}(x, [a_{n1}, b_{n1}, c_{n1}, a_{n2}, b_{n2}, c_{n2}]), \\ \text{where } &a_{n1} < a_{n2}, b_{n1} < b_{n2}, c_{n1} < c_{n2}, \\ \mu_1(x) &= \max\left(\min\left(\frac{x - a_1}{b_1 - a_1}, \frac{c_1 - x}{c_1 - b_1}\right), 0\right), \\ \mu_2(x) &= \max\left(\min\left(\frac{x - a_2}{b_2 - a_2}, \frac{c_2 - x}{c_2 - b_2}\right), 0\right), \\ \bar{\mu}(x) &= \max(\mu_1(x), \mu_2(x)) \forall_x (b_{n1}, b_{n2}), \\ \underline{\mu}(x) &= 1 \forall_x f(b_{n1}, b_{n2}), \\ \underline{\mu}(x) &= \min(\mu_1(x), \mu_2(x)). \end{aligned} \tag{10}$$

In Figs. 7 and 8, we can find the graphical description of the interval type-2 triangular and trapezoidal membership functions, respectively.

The representation of the interval type-2 trapezoidal membership functions (itrapatyp2) of the fuzzy controller consists of 8 parameters  $a_{n1}$ ,  $b_{n1}$ ,  $c_{n1}$ ,  $d_{n1}$ ,  $a_{n2}$ ,  $b_{n2}$ ,  $c_{n2}$  and  $d_{n2}$ , where  $a_{n1} < a_{n2}$ ,  $b_{n1} < b_{n2}$ ,  $c_{n1} < c_{n2}$  and  $d_{n1} < d_{n2}$  and this is shown as follows:

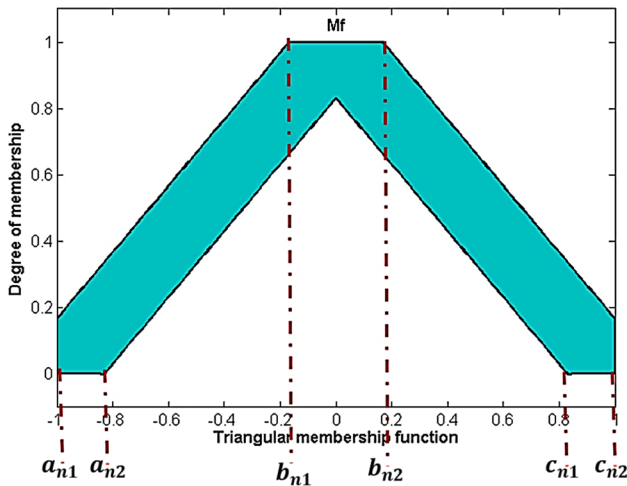


Fig. 7 Interval type-2 triangular membership function

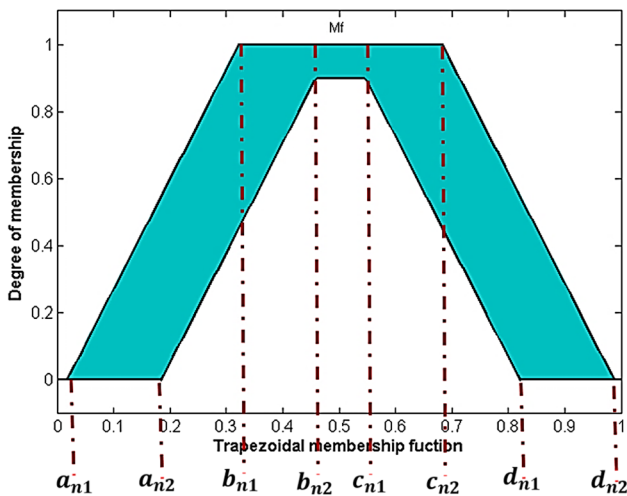


Fig. 8 Interval type-2 trapezoidal membership function

$$\mu(x) = \text{itrapatype2}(x, [a_{n1}, b_{n1}, c_{n1}, d_{n1}, a_{n2}, b_{n2}, c_{n2}, d_{n2}]),$$

where  $a_{n1} < a_{n2}, b_{n1} < b_{n2}, c_{n1} < c_{n2}, d_{n1} < d_{n2},$

$$\mu_1(x) = \max\left(\min\left(\frac{x - a_{n1}}{b_{n1} - a_{n1}}, 1, \frac{d_{n1} - x}{d_{n1} - c_{n1}}\right), 0\right),$$

$$\mu_2(x) = \max\left(\min\left(\frac{x - a_{n2}}{b_{n2} - a_{n2}}, 1, \frac{d_{n2} - x}{d_{n2} - c_{n2}}\right), 0\right),$$

$$\bar{\mu}(x) = \begin{cases} \max(\mu_1(x), \mu_2(x)) & \forall x \notin (b1, c2) \\ 1 & \forall x \in (b1, c2), \end{cases}$$

$$\underline{\mu}(x) = \min(\alpha, \min(\mu_1(x), \mu_2(x))). \tag{11}$$

### 4 Problem Description (Autonomous Mobile Robot)

To test the proposed methodology, we consider the case of controlling an autonomous mobile robot. For this particular case, the main function of the fuzzy controller is to provide

control of the motors to minimize the error in an established trajectory [29, 30].

The robot design for the case study can be found in Fig. 9, which consists of 2 wheels that move by means of motors and a passive wheel to give stability to the robot during the motion. The operation of the robot is determined by means of the following equations [31, 32]:

$$M(q)\dot{v} + C(q, \dot{q})v + D_v = \tau + P(\tau), \tag{12}$$

$$\dot{q} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix}, \tag{13}$$

where [33]  $q = (x, y, \theta)^T$  describes the position of the robot;  $v = (v, w)^T$  represents the linear and angular velocities;  $\tau = (\tau_1, \tau_2)$  represents the torque that is applied to the wheels;

$P \in R^2$  is the disturbance vector;  $M(q) \in R^{2 \times 2}$  represents a positive inertia matrix;  $C(q, \dot{q})v$  describes the vector of centripetal and Coriolis forces, and  $D \in R^{2 \times 2}$  is a diagonal positive defined damping matrix.

The fuzzy controller of the mobile robot is formed by two input variables labeled as the errors of the linear and angular velocities, granulated into three membership functions, two trapezoidal for the edges and one triangular for the center labeled as negative, zero and positive, respectively. The output variables are called *torque1* and *torque2*, which are granulated into three triangular membership functions labeled: negative, zero and positive [34, 35]. Below we have a graphic description of the inputs and outputs that form the interval type-2 fuzzy controller of the mobile autonomous robot that, as mentioned above, which are granulated with triangular and trapezoidal membership functions. Figures 10, 11, 12 and 13 illustrate the corresponding type-2 membership functions.

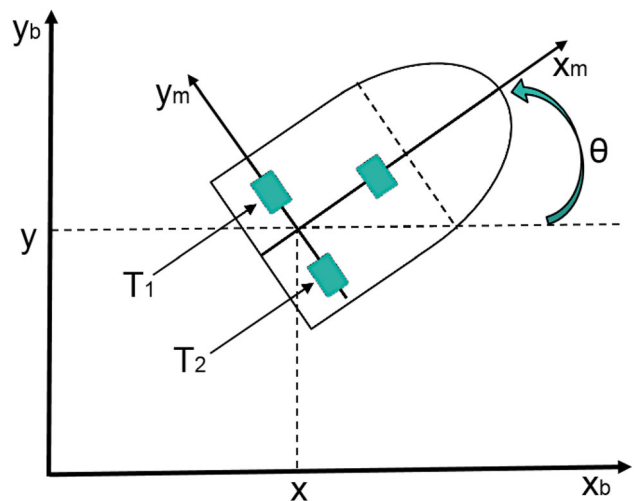


Fig. 9 Mobile robot model

The design of the fuzzy rules of the autonomous mobile robot controller enables controlling the movement of the robot following the established trajectory, so that the movement is uniform and continuous. The fuzzy rules are listed below:

1. If (ErrorLV is negative) and (ErrorAV is negative) then (Torque1 is negative) and (Torque2 is negative),
2. If (ErrorLV is negative) and (ErrorAV is zero) then (Torque1 is negative) and (Torque2 is zero),
3. If (ErrorLV is negative) and (ErrorAV is positive) then (Torque1 is negative) and (Torque2 is positive),
4. If (ErrorLV is zero) and (ErrorAV is negative) then (Torque1 is zero) and (Torque2 is negative),
5. If (ErrorLV is zero) and (ErrorAV is zero) then (Torque1 is zero) and (Torque2 is zero),
6. If (ErrorLV is zero) and (ErrorAV is positive) then (Torque1 is zero) and (Torque2 is positive),
7. If (ErrorLV is positive) and (ErrorAV is negative) then (Torque1 is positive) and (Torque2 is negative),
8. If (ErrorLV is positive) and (ErrorAV is zero) then (Torque1 is positive) and (Torque2 is zero),
9. If (ErrorLV is positive) and (ErrorAV is positive) then (Torque1 is positive) and (Torque2 is positive).

These fuzzy rules are basically deciding the particular actions on the left and right wheels according to the position of the robot with respect to the desired trajectory.

### 5 Simulation Results

In this section, the results obtained from the fuzzy controller optimization of the autonomous mobile robot are presented. The methodology consists on using a meta-heuristic algorithm to generate a vector of the necessary parameters to form the membership functions of the

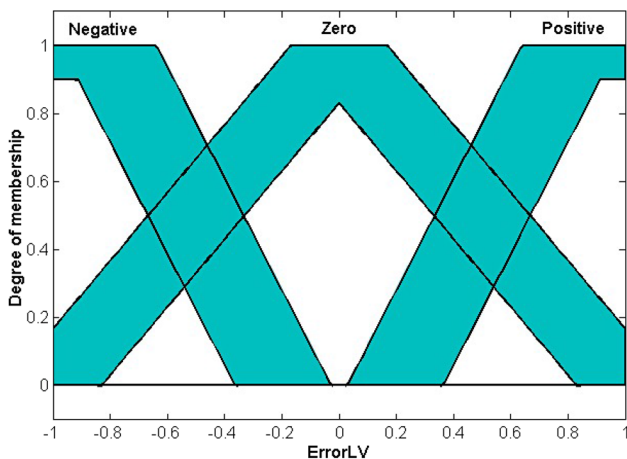


Fig. 10 Membership functions of the linear velocity error

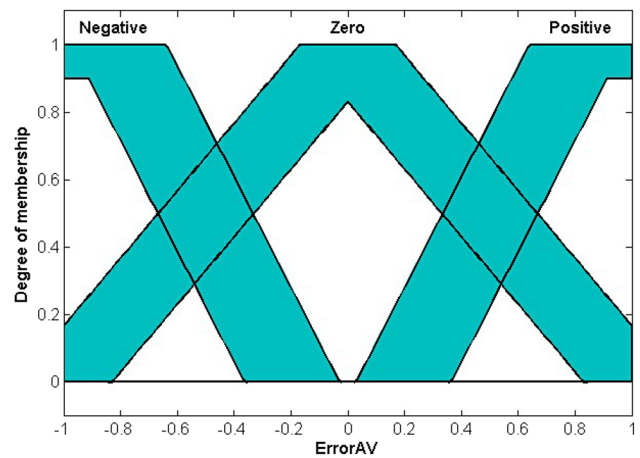


Fig. 11 Membership functions of the angular velocity error

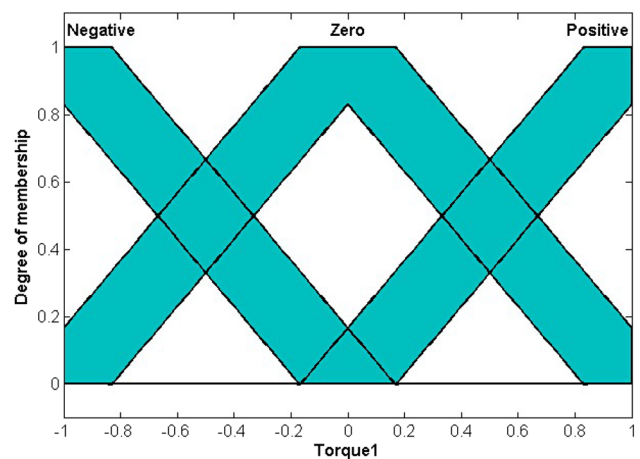


Fig. 12 Membership functions of Torque 1

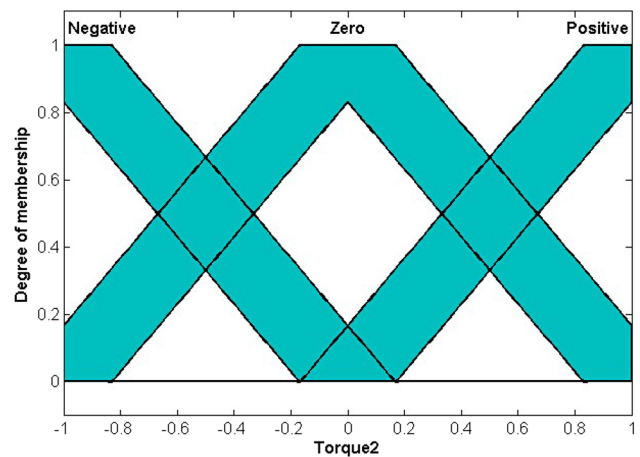


Fig. 13 Membership functions of Torque 2

interval type-2 fuzzy controller that is optimized. For this specific case, the metaheuristics are the galactic swarm optimization and firefly algorithm and their variants with dynamic adaptation of parameters using type-1 and interval



type-2 fuzzy systems. Table 1 shows the parameters used in galactic swarm optimization and firefly algorithm to perform the optimization of the fuzzy controller.

To measure the performance of the algorithms for the optimization of the fuzzy controller, the mean squared error (MSE) is used. Its equation is described below:

$$MSE = \frac{1}{n} \sum_{i=1}^n (X_i - Y_i)^2, \tag{14}$$

where  $X_i$  is the reference value at time  $I$ ; the reference values are given in [32];  $Y_i$  is the value produced by the system at time  $I$ , and  $n$  is the number of samples considered in the test.

Table 2 shows the performed experiments to obtain the best optimized fuzzy system with the FA, where it can be observed that the best error found is of  $9.64 \times 10^{-02}$  and an average of  $3.20 \times 10^{+00}$  was obtained.

Table 3 shows the performed experiments to obtain the best optimized fuzzy system with the fuzzy firefly algorithm, where it can be observed that the best error found is of  $9.93 \times 10^{-01}$  and an average of  $1.67 \times 10^{+00}$  was obtained.

Table 4 shows the performed experiments to obtain the best optimized fuzzy system with the interval Type-2 fuzzy firefly algorithm, where it can be observed that the best error found is of  $9.64 \times 10^{-02}$  and an average of  $1.06 \times 10^{+00}$  is obtained.

Table 5 shows the performed experiments to obtain the best optimized fuzzy system with the GSO algorithm, where it can be observed that the best error found is of  $5.05 \times 10^{-03}$  and an average of  $1.17 \times 10^{+00}$  was obtained.

Table 6 shows the performed experiments to obtain the best optimized fuzzy system with the FGSO algorithm, where it can be observed that the best error found is of  $5.78 \times 10^{-06}$  and an average of  $8.84 \times 10^{-01}$  was achieved.

Table 7 shows the performed experiments to obtain the best optimized the interval type-2 fuzzy system with GSO algorithm, where it can be observed that the best error

**Table 1** Parameters of the GSO and FA algorithms

Parameter	GSO	Parameter	FA
Population	10	Population	50
Subpopulation	5	Iteration	750
Iteration 1	5	–	–
Iteration 2	10	–	–
$C_1$ and $C_2$	2.5	–	–
$C_3$ and $C_4$	Dynamic	–	–

**Table 2** Results for the FA

Experiment	MSE	Experiment	MSE
1	$2.52 \times 10^{-01}$	16	$7.06 \times 10^{+00}$
2	$3.28 \times 10^{-01}$	17	$7.08 \times 10^{+00}$
3	$5.72 \times 10^{-01}$	18	$9.49 \times 10^{+00}$
4	$7.06 \times 10^{+00}$	19	$9.75 \times 10^{+00}$
5	$4.42 \times 10^{+00}$	20	$1.01 \times 10^{+00}$
6	$9.49 \times 10^{+00}$	21	$1.26 \times 10^{+00}$
7	$9.64 \times 10^{-02}$	22	$2.68 \times 10^{+00}$
8	$2.52 \times 10^{-01}$	23	$3.13 \times 10^{+00}$
9	$3.27 \times 10^{-01}$	24	$3.15 \times 10^{+00}$
10	$5.25 \times 10^{-01}$	25	$1.24 \times 12^{+00}$
11	$5.72 \times 10^{-01}$	26	$7.35 \times 10^{+00}$
12	$1.13 \times 10^{+00}$	27	$1.21 \times 10^{+00}$
13	$3.84 \times 10^{+00}$	28	$1.34 \times 10^{+00}$
14	$4.42 \times 10^{+00}$	29	$1.16 \times 10^{+00}$
15	$4.43 \times 10^{+00}$	30	$1.29 \times 10^{+00}$

**Table 3** Results for the fuzzy FA

Experiment	MSE	Experiment	MSE
1	$1.67 \times 10^{+00}$	16	$6.63 \times 10^{-01}$
2	$1.76 \times 10^{+00}$	17	$8.83 \times 10^{-01}$
3	$3.24 \times 10^{-01}$	18	$3.06 \times 10^{-01}$
4	$7.45 \times 10^{-01}$	19	$8.82 \times 10^{-01}$
5	$9.70 \times 10^{-01}$	20	$3.44 \times 10^{-01}$
6	$3.06 \times 10^{-01}$	21	$9.98 \times 10^{-01}$
7	$1.37 \times 10^{+00}$	22	$3.22 \times 10^{-01}$
8	$9.80 \times 10^{-01}$	23	$3.31 \times 10^{-01}$
9	$1.87 \times 10^{+00}$	24	$4.43 \times 10^{-01}$
10	$1.89 \times 10^{+00}$	25	$8.34 \times 10^{-01}$
11	$3.64 \times 10^{-01}$	26	$9.22 \times 10^{-01}$
12	$3.53 \times 10^{-01}$	27	$7.73 \times 10^{-01}$
13	$3.06 \times 10^{-01}$	28	$3.46 \times 10^{-01}$
14	$8.79 \times 10^{-01}$	29	$9.93 \times 10^{-01}$
15	$3.06 \times 10^{-01}$	30	$3.06 \times 10^{-01}$

found is of  $1.39 \times 10^{-05}$  and an average of  $8.35 \times 10^{-01}$  was achieved.

Table 8 shows the best, the worst, the average and standard deviations after 30 executions with the FA and its variants after having performed the fuzzy controller optimization.

Table 9 shows the best, the worst, the average and standard deviations after 30 executions with the GSO

**Table 4** Results for the interval Type-2 fuzzy FA

Experiment	MSE	Experiment	MSE
1	$1.27 \times 10^{+00}$	16	$1.06 \times 10^{+00}$
2	$1.90 \times 10^{+00}$	17	$3.28 \times 10^{-01}$
3	$5.72 \times 10^{-01}$	18	$1.06 \times 10^{+00}$
4	$1.60 \times 10^{+00}$	19	$8.75 \times 10^{-01}$
5	$1.27 \times 10^{+00}$	20	$7.35 \times 10^{-01}$
6	$4.04 \times 10^{-01}$	21	$9.12 \times 10^{-01}$
7	$1.14 \times 10^{+00}$	22	$7.47 \times 10^{-01}$
8	$9.64 \times 10^{-02}$	23	$9.34 \times 10^{-01}$
9	$1.06 \times 10^{+00}$	24	$8.82 \times 10^{-01}$
10	$1.09 \times 10^{+00}$	25	$7.45 \times 10^{-01}$
11	$1.06 \times 10^{+00}$	26	$1.12 \times 10^{-01}$
12	$2.53 \times 10^{-01}$	27	$8.76 \times 10^{-01}$
13	$1.06 \times 10^{+00}$	28	$9.98 \times 10^{-01}$
14	$9.70 \times 10^{-01}$	29	$7.45 \times 10^{-01}$
15	$5.25 \times 10^{-01}$	30	$3.38 \times 10^{-01}$

**Table 6** Results for fuzzy galactic swarm optimization (FGSO1)

Experiment	MSE	Experiment	MSE
1	$4.57 \times 10^{-01}$	16	$2.63 \times 10^{-05}$
2	$1.60 \times 10^{-01}$	17	$3.44 \times 10^{-01}$
3	$1.68 \times 10^{-01}$	18	$3.90 \times 10^{-02}$
4	$1.40 \times 10^{-01}$	19	$5.84 \times 10^{-01}$
5	$5.78 \times 10^{-06}$	20	$4.94 \times 10^{-01}$
6	$3.63 \times 10^{-02}$	21	$6.69 \times 10^{-01}$
7	$1.59 \times 10^{-01}$	22	$5.19 \times 10^{-01}$
8	$6.08 \times 10^{-01}$	23	$5.46 \times 10^{-02}$
9	$2.33 \times 10^{-01}$	24	$2.23 \times 10^{-02}$
10	$3.08 \times 10^{-01}$	25	$2.43 \times 10^{-02}$
11	$1.45 \times 10^{-01}$	26	$1.19 \times 10^{-01}$
12	$1.73 \times 10^{-01}$	27	$8.84 \times 10^{-01}$
13	$1.25 \times 10^{-01}$	28	$8.41 \times 10^{-01}$
14	$3.74 \times 10^{-01}$	29	$1.61 \times 10^{-01}$
15	$8.11 \times 10^{-02}$	30	$1.82 \times 10^{-01}$

**Table 5** Results for galactic swarm optimization (GSO)

Experiment	MSE	Experiment	MSE
1	$8.65 \times 10^{-01}$	16	$3.41 \times 10^{-01}$
2	$7.74 \times 10^{-02}$	17	$7.14 \times 10^{-02}$
3	$2.32 \times 10^{-01}$	18	$7.40 \times 10^{-01}$
4	$1.20 \times 10^{-01}$	19	$1.44 \times 10^{+00}$
5	$2.23 \times 10^{-01}$	20	$9.20 \times 10^{-01}$
6	$1.96 \times 10^{+00}$	21	$1.19 \times 10^{+00}$
7	$1.63 \times 10^{-02}$	22	$7.03 \times 10^{+00}$
8	$5.93 \times 10^{-03}$	23	$4.71 \times 10^{+00}$
9	$4.67 \times 10^{-01}$	24	$5.58 \times 10^{+00}$
10	$2.88 \times 10^{-01}$	25	$8.64 \times 10^{-02}$
11	$5.05 \times 10^{-03}$	26	$8.46 \times 10^{-01}$
12	$6.97 \times 10^{-01}$	27	$3.54 \times 10^{+00}$
13	$2.91 \times 10^{-02}$	28	$3.52 \times 10^{-01}$
14	$6.65 \times 10^{-03}$	29	$3.89 \times 10^{-01}$
15	$3.35 \times 10^{-01}$	30	$2.58 \times 10^{+00}$

**Table 7** Results for interval Type-2 fuzzy galactic swarm optimization (FGSO IT2)

Experiment	MSE	Experiment	MSE
1	$7.87 \times 10^{-01}$	16	$2.67 \times 10^{-01}$
2	$3.53 \times 10^{-01}$	17	$4.16 \times 10^{-01}$
3	$8.50 \times 10^{-03}$	18	$8.35 \times 10^{-01}$
4	$3.90 \times 10^{-03}$	19	$3.80 \times 10^{-01}$
5	$6.28 \times 10^{-01}$	20	$6.60 \times 10^{-01}$
6	$1.05 \times 10^{-02}$	21	$1.39 \times 10^{-05}$
7	$8.29 \times 10^{-02}$	22	$7.40 \times 10^{-01}$
8	$1.06 \times 10^{-01}$	23	$1.19 \times 10^{-02}$
9	$6.83 \times 10^{-01}$	24	$1.38 \times 10^{-01}$
10	$8.18 \times 10^{-01}$	25	$1.38 \times 10^{-01}$
11	$1.06 \times 10^{-01}$	26	$3.40 \times 10^{-01}$
12	$3.33 \times 10^{-01}$	27	$2.29 \times 10^{-01}$
13	$5.84 \times 10^{-01}$	28	$3.13 \times 10^{-01}$
14	$6.74 \times 10^{-01}$	29	$1.25 \times 10^{-01}$
15	$1.14 \times 10^{-02}$	30	$1.58 \times 10^{-01}$

algorithm and its variants after having performed the fuzzy controller optimization.

Finally, the results obtained with the methods are presented in Figs. 14, 15, 16 and 17, respectively, where one can note the differences between the desired trajectory and the real one generated by the autonomous mobile robot with the type-2 fuzzy logic controller.

### 5.1 Statistical Comparison

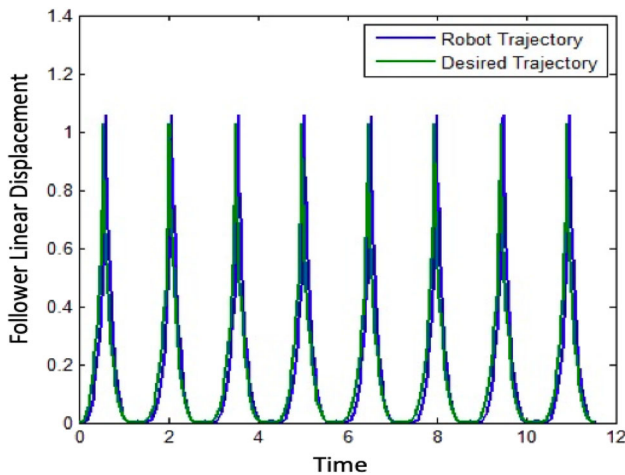
In order to validate the performance of the metaheuristic algorithms used to test the proposed optimization methodology of membership functions parameters of the mobile robot fuzzy controller, a statistical comparison is made to find evidence that the algorithms have performed

**Table 8** Best, worst and average for the FA and its variants

	FA	FA_T1	FA_T2
Best	$9.64 \times 10^{-02}$	$9.93 \times 10^{-01}$	$9.64 \times 10^{-02}$
Worst	$9.75 \times 10^{+00}$	$1.67 \times 10^{+00}$	$1.06 \times 10^{+00}$
Average	$3.20 \times 10^{+00}$	$7.81 \times 10^{-01}$	$8.54 \times 10^{-01}$
Standard deviation	$3.14 \times 10^{+00}$	$5.03 \times 10^{-01}$	$4.09 \times 10^{-01}$

**Table 9** Best, worst and average for the GSO algorithm and its variants

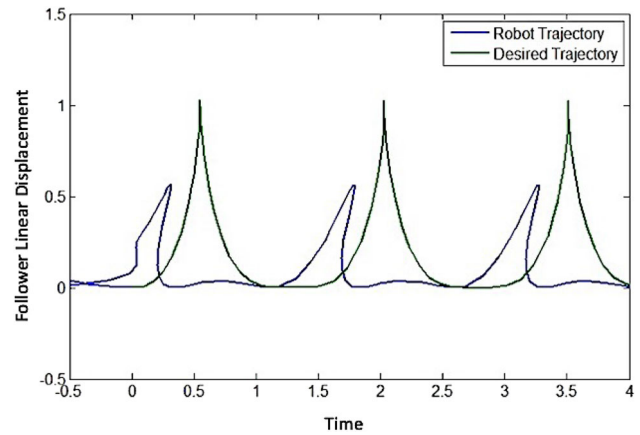
	GSO	FGSO1	FGSO IT2
Best	$5.05 \times 10^{-03}$	$5.78 \times 10^{-06}$	$1.39 \times 10^{-05}$
Worst	$7.04 \times 10^{+00}$	$8.84 \times 10^{-01}$	$8.35 \times 10^{-01}$
Average	$1.10 \times 10^{+00}$	$2.70 \times 10^{-01}$	$3.34 \times 10^{-01}$
Standard deviation	$1.77 \times 10^{+00}$	$2.48 \times 10^{-01}$	$2.81 \times 10^{-01}$



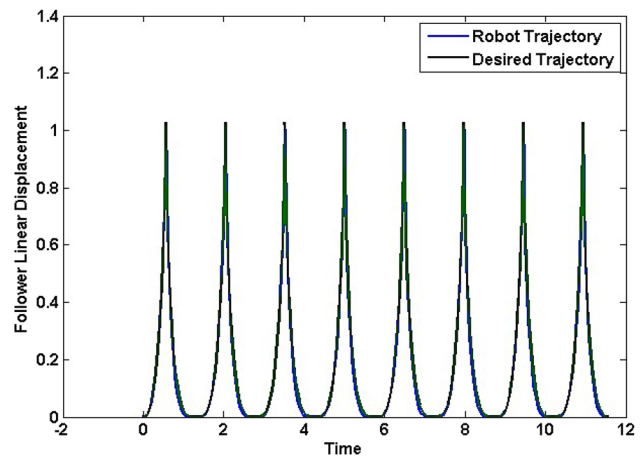
**Fig. 14** The best result ( $9.64 \times 10^{-02}$ ) obtained with firefly algorithm (FA) and its variants

well in the case study of the autonomous mobile robot plant. Table 10 shows the representation of the values used in the statistical test, to determine which method produced better results in the optimization of membership function parameters, generating a data vector, which helps to improve the performance of the fuzzy controller for the autonomous robot.

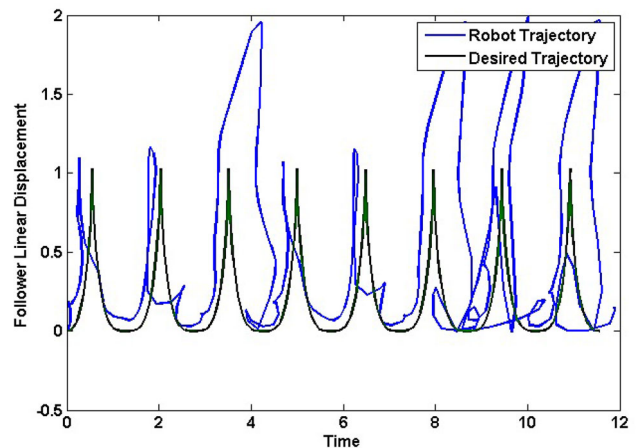
In Table 10, the null hypothesis ( $H_0$ ) states that the average MSE error for the GSO algorithm and its variants ( $\mu_1$ ) is greater than or equal to the average MSE error for FA and its variants ( $\mu_2$ ). The alternative hypothesis ( $H_a$ ) states that the average MSE error for the GSO algorithm



**Fig. 15** The worst result ( $9.75 \times 10^{+00}$ ) obtained with firefly algorithm (FA) and its variants



**Fig. 16** The best result ( $5.78 \times 10^{-06}$ ) obtain with the GSO algorithm and its variants



**Fig. 17** The worst result ( $7.04 \times 10^{+00}$ ) obtain with the GSO algorithm and its variants

**Table 10** Statistical test parameters [24, 35]

Parameter	Value
$H_0$	$\mu_1 \geq \mu_2$
$H_a$	$\mu_1 < \mu_2$ (claim)
Level of significance	95%
$A$	0.05
Critical value	- 1.645

and its variants ( $\mu_1$ ) is less than the average MSE error for FA and its variants ( $\mu_2$ ).

Table 11 shows the averages, standard deviations and values of  $z$  for each of the comparisons made, where the GSO algorithm and its variants manage to obtain significant evidence against the FA and its variants. As a consequence  $H_0$  is rejected and  $H_a$  is accepted with a 95% of the level of significance and a rejection zone for values lower than - 1.645.

As mentioned above, the metaheuristic algorithms generate satisfactory results in optimization problems, and in this paper we optimized a type-2 fuzzy controller which controls autonomous robot navigation, and then the results obtained in the experimentation are described. This experimentation is made with two optimization methods for the type-2 fuzzy controller to make a comparison of the methods and thus, discover which one is the best. As can be noted in Table 11, the method that produces the best results according to the hypothesis testing is the galactic swarm optimization, since the firefly algorithm gives a higher difference error between the desired trajectory and the actual trajectory generated by the robot. This algorithm has shown in other optimization problems to give good results as explained in [36, 37], although on the other hand, the galactic swarm optimization proved to be better in terms of the solution for this problem.

In addition, we also used the nonparametric Friedman test to be even more confident about the validity of the results. The Friedman test [38–40] is a nonparametric test

**Table 11** Results of the Z-test for the GSO algorithm and the FA

GSO		FA		Z value
Average	Std	Average	Std	
$1.10 \times 10^{+00}$	$1.77 \times 10^{+00}$	$3.20 \times 10^{+00}$	$3.14 \times 10^{+00}$	$-4.85 \times 10^{+01}$
FGSO		FA_T1		
$2.70 \times 10^{-01}$	$2.48 \times 10^{-01}$	$7.81 \times 10^{-01}$	$5.03 \times 10^{-01}$	
FGSO IT2		FA_T2		
$3.34 \times 10^{-01}$	$2.81 \times 10^{-01}$	$8.54 \times 10^{-01}$	$4.09 \times 10^{-01}$	

**Table 12** Friedman test in comparing GSO versus FA

GSO vs FA	
Test statistic	p-value
6.53333	0.01059
$Q = \frac{12n}{k(k+1)} \left[ \sum_j R_j^2 - \frac{k(k+1)^2}{3} \right]$	

to compare multiple algorithms to find the best algorithm. First, if we compare the original GSO and FA methods, we find that there is a significant advantage of GSO with a  $p$  value of 0.01 (see Table 12). Also, if we compare FGSO and FA\_T1, we find significant advantage of FGSO with a  $p$  value of 0.00006 (see Table 13). Finally, we can compare FGSO IT2 versus FA\_T2 and we find significant advantage of FGSO IT2 with a  $p$  value of 0.00006 (see Table 14).

In summary, we can state that there is sufficient statistical evidence to say that the galactic swarm optimization (and its variants) outperforms the firefly algorithm.

**Table 13** Friedman test in comparing FGSO versus FA\_T1

FGSO vs FA_T1	
Test statistic	p-value
16.13333	0.00006
$Q = \frac{12n}{k(k+1)} \left[ \sum_j R_j^2 - \frac{k(k+1)^2}{3} \right]$	
$Q = 0.066666666 * 4292.0 - 270$	
$Q = 16.13333$	

**Table 14** Friedman test in comparing FGSO IT2 versus FA\_T2

FGSO IT2 vs FA_T2	
Test statistic	<i>p</i> -value
16.13333	0.00006
$Q = \frac{12n}{k(K+1)} \left[ \sum_j R_j^2 - \frac{k(k+1)^2}{3} \right]$	
$Q = 0.06666666 * 4148.0 - 270$	
$Q = 16.13333$	

### 6 Discussion

In the state of the art, we can find works related to the optimization of membership functions, such as in [41] where a comparative study of type-2 fuzzy particle swarm, bee colony and bat algorithms in optimization of fuzzy controllers is presented. This study makes a comparison between three very promising methods in optimization of fuzzy controllers, where their results for each of the algorithms show the best, worst, average, standard deviation. So having these results we can compare them with the results obtained in this proposal, where we can discuss which method has achieved better performance in the optimization of fuzzy controllers. In Table 15, the results of bee colony optimization are illustrated versus results of Table 8 that represents firefly algorithm (FA), and it is observed that the best value found by the original bee colony optimization (BCO) algorithm, the best with type-1 fuzzy logic (BCO\_T1), type-2 fuzzy logic (BCO\_T2), but on average and standard deviation the FA showed that its performance is better in optimizing membership functions.

Regarding Table 16 which represents particle swarm optimization (PSO particle swarm optimization with type-1 fuzzy logic (PSO\_T1), particle swarm optimization with type-2 fuzzy logic (PSO\_T2), it is observed that the results were better for PSO than for the FA in Table 8, but it must be taken into account that the PSO with inertia weight parameters, position update that helps improve the movement of the particles to find the best overall, and the

**Table 16** MSE results of PSO

MSE [38]	PSO	PSO_T1	PSO_T2
Best	$1.39 \times 10^{-01}$	$4.41 \times 10^{-03}$	$1.90 \times 10^{-04}$
Worst	$6.67 \times 10^{+00}$	$1.51 \times 10^{+00}$	$3.3 \times 10^{-01}$
Average	$2.41 \times 10^{+00}$	$3.04 \times 10^{-01}$	$3.22 \times 10^{-02}$
Standard deviation	$2.97 \times 10^{+00}$	$3.19 \times 10^{-01}$	$7.11 \times 10^{-02}$

**Table 17** Results of the bat algorithm

MSE [38]	BA	BA_T1	BA_T2
Best	$4 \times 10^{-03}$	$1.47 \times 10^{-02}$	$1.64 \times 10^{+00}$
Worst	$9.31 \times 10^{+00}$	$97.98 \times 10^{+00}$	$34.31 \times 10^{+00}$
Average	$4.02 \times 10^{+00}$	$36.80 \times 10^{+00}$	$20.47 \times 10^{+00}$
Standard deviation	$2.90 \times 10^{+00}$	$34.57 \times 10^{+00}$	$12.25 \times 10^{+00}$

traditional FA only uses a next position and a Gaussian distribution.

In Table 17, the bat algorithm shows results that are inferior in its optimization performance as shown in Table 17 versus the FA of Table 8.

Table 9 shows the results obtained with GSO, which are better when compared to Tables 15 and 17. As it is observed, in general the proposed algorithms turn out to have better performance than those found in the literature. We have to mention that there are other related articles, but they use the RMSE metric and not the MSE that is used in this proposal.

Finally, we show in Table 18 a summary of all the above-mentioned methods, and we can note that the best fuzzy controller (highlighted in bold) is found by FGSO IT2 with an error of  $1.39 \times 10^{-05}$ . We can also notice that PSO\_T2 achieves the second best fuzzy controller with an error of  $1.90 \times 10^{-04}$ . From the optimal fuzzy control design point of view this is very important because the goal is finding the best possible controller. However, statistically speaking on average there is no significant difference

**Table 15** Results of performance the BCO

MSE [38]	BCO	BCO_T1	BCO_T2
Best	$8.84 \times 10^{-03}$	$1.50 \times 10^{-03}$	$2.80 \times 10^{-03}$
Worst	$106.81 \times 10^{+00}$	$61.91 \times 10^{+00}$	$65.46 \times 10^{+00}$
Average	$14.61 \times 10^{+00}$	$9.83 \times 10^{+00}$	$9.10 \times 10^{+00}$
Standard deviation	$23.34 \times 10^{+00}$	$14.69 \times 10^{+00}$	$16.67 \times 10^{+00}$

**Table 18** Summary of results among the proposed methods and other algorithms in the literature

MSE	Best	Worst	Average	Standard deviation
FA	$9.64 \times 10^{-02}$	$9.75 \times 10^{+00}$	$3.20 \times 10^{+00}$	$3.14 \times 10^{+00}$
GSO	$5.05 \times 10^{-03}$	$7.04 \times 10^{+00}$	$1.67 \times 10^{+00}$	$8.84 \times 10^{-01}$
BA	$4 \times 10^{-03}$	$9.31 \times 10^{+00}$	$4.02 \times 10^{+00}$	$2.90 \times 10^{+00}$
BCO	$8.84 \times 10^{-03}$	$106.81 \times 10^{+00}$	$14.61 \times 10^{+00}$	$23.34 \times 10^{+00}$
PSO	$1.39 \times 10^{-01}$	$6.67 \times 10^{+00}$	$2.41 \times 10^{+00}$	$2.97 \times 10^{+00}$
FA_T1	$9.93 \times 10^{-01}$	$1.67 \times 10^{+00}$	$7.81 \times 10^{-01}$	$5.03 \times 10^{-01}$
FGSO1	$3.14 \times 10^{+00}$	$1.77 \times 10^{+00}$	$5.03 \times 10^{-01}$	$2.48 \times 10^{-01}$
BA_T1	$1.47 \times 10^{-02}$	$97.98 \times 10^{+00}$	$36.80 \times 10^{+00}$	$34.57 \times 10^{+00}$
BCO_T1	$1.50 \times 10^{-03}$	$61.91 \times 10^{+00}$	$9.83 \times 10^{+00}$	$14.69 \times 10^{+00}$
PSO_T1	$4.41 \times 10^{-03}$	$1.51 \times 10^{+00}$	$3.04 \times 10^{-01}$	$3.19 \times 10^{-01}$
FA_T2	$9.64 \times 10^{-02}$	$1.06 \times 10^{+00}$	$8.54 \times 10^{-01}$	$4.09 \times 10^{-01}$
FGSO IT2	$1.39 \times 10^{-05}$	$8.35 \times 10^{-01}$	$3.34 \times 10^{-01}$	$2.81 \times 10^{-01}$
BA_T2	$1.64 \times 10^{+00}$	$34.31 \times 10^{+00}$	$20.47 \times 10^{+00}$	$12.25 \times 10^{+00}$
BCO_T2	$2.80 \times 10^{-03}$	$65.46 \times 10^{+00}$	$9.10 \times 10^{+00}$	$16.67 \times 10^{+00}$
PSO_T2	$1.90 \times 10^{-04}$	$3.3 \times 10^{-01}$	$3.22 \times 10^{-02}$	$7.11 \times 10^{-02}$

between the type-2 fuzzy GSO variant and the type-2 fuzzy PSO algorithm.

## 7 Conclusions

Fuzzy controllers are being widely used today, as they can solve problems that were previously almost impossible to deal with. As a conclusion of this paper it is explained that the results that originated by the parameter optimization of the type-2 fuzzy robot controller membership functions were very satisfactory. In addition, the obtained controller values proved to be very close to the robot's desired trajectory, observing with this, that the performance of a fuzzy controller can be improved if a metaheuristic optimization is applied. In this case, we used two metaheuristics that proved to be good in parameter optimization of the membership functions.

The experimentation was carried out with the original GSA and FA metaheuristics and their variants using type-1 and interval type-2 fuzzy systems for dynamic parameter adjustment. In order to compare the variants, a statistical test was performed between the original GSO algorithm and the FA and their variants using type-1 and interval type-2 fuzzy systems, where the GSO algorithm achieved significant improvements when compared to the FA based on the results shown in Table 11.

We can conclude that the parameter adjustment using fuzzy logic in the GSO and FA algorithms applied to fuzzy control is a good choice since competitive results were

obtained with these methods. In this regard, we plan to continue investigating in this line of research and depending on the behavior of the algorithms and the use of type-1 and type-2 fuzzy logic, we will consider other types of problems.

As future work, different variations of the fuzzy systems for the dynamic adjustment of the parameters in the metaheuristic algorithms will be considered and applied to different control problems.

It is also planned to perform the dynamic tuning of the parameters with generalized type-2 fuzzy systems and in the same way to extend the controller to a generalized type-2 fuzzy controller [42, 43]. In addition, a study of the controllers by adding different levels of disturbances to the plants could be done to measure the performance of the proposed method under different situations.

## References

1. Zadeh, L.A.: Fuzzy sets. *Inf. Control* **8**, 338–353 (1965)
2. Zadeh, L.A.: Fuzzy logic. *Computer* **21**, 83–93 (1988)
3. Zadeh, L.A.: Fuzzy logic computing with words. *IEEE Trans. Fuzzy Syst.* **4**(2), 103–111 (1996)
4. Liang, Q., Mendel, J.M.: Interval type-2 fuzzy logic systems: theory and design. *IEEE Trans. Fuzzy Syst.* **8**(5), 535–550 (2000)
5. Du, Z., Kao, Y., Zhao, X.: An input delay approach to interval type-2 fuzzy exponential stabilization for nonlinear unreliable networked sampled-data control systems. *IEEE Trans. Syst. Man Cybern. Syst.* (2019). <https://doi.org/10.1109/TSMC.2019.2930473>

6. Qu, Z., Zhang, Z., Du, Z., Peng, M.: Interval type-2 fuzzy sampled-data optimal control for nonlinear systems with multiple conditions. *Int. J. Fuzzy Syst.* **21**(5), 1480–1496 (2019)
7. Du, Z., Kao, Y., Park, J.H.: New results for sampled-data control of interval type-2 fuzzy nonlinear systems. *J. Franklin Inst.* **357**(1), 121–141 (2020)
8. Du, Zhenbin, Yan, Zhenzheng, Zhao, Zhen: Interval type-2 fuzzy tracking control for nonlinear systems via sampled-data controller. *Fuzzy Sets Syst.* **356**, 92–112 (2019)
9. Mendel, J.M.: Advances in type-2 fuzzy sets and systems. *Inf. Sci.* **177**(1), 84–110 (2007)
10. Hagrass, H.: Type-2 FLCs: a new generation of fuzzy controllers. *IEEE Comput. Intell. Mag.* **2**(1), 30–43 (2007)
11. Lagunes, M. L., Castillo, O., Soria, J.: Methodology for the optimization of a fuzzy controller using a bio-inspired algorithm. In: *Proceedings of the North American Fuzzy Information Processing Society Annual Conference, Cancun, Mexico, 16 October 2017*; Springer, Cham, Switzerland, Vol. 648, pp. 131–137 (2018)
12. Lagunes, M.L., Castillo, O., Soria, J.: Optimization of membership function parameters for fuzzy controllers of an autonomous mobile robot using the firefly algorithm. In: *Fuzzy Logic Augmentation of Neural and Optimization Algorithms: Theoretical Aspects and Real Applications*. Springer, Berlin, Germany, pp. 199–206 (2018)
13. Lagunes, M.L., Castillo, O., Valdez, F., Soria, J., Melin, P.: Parameter optimization for membership functions of type-2 fuzzy controllers for autonomous mobile robots using the firefly algorithm. In: *Proceedings of the North American Fuzzy Information Processing Society Annual Conference, Cancun, Mexico, 16 October 2017*; Springer, Berlin, Germany. IEEE, Piscataway, NJ, USA, pp. 569–579 (2018)
14. Bernal, E., et al.: Optimization of fuzzy controller using galactic swarm optimization with type-2 fuzzy dynamic parameter adjustment. *Axioms* **8**(1), 26 (2019)
15. Bernal, E., Castillo, O., Soria, J., Valdez, F.: Fuzzy galactic swarm optimization with dynamic adjustment of parameters based on fuzzy logic. *Metaheuristics*. 1–19 (2018)
16. Wagner, C., Hagrass, H.: A genetic algorithm based architecture for evolving type-2 fuzzy logic controllers for real world autonomous mobile robots. In: *2007 IEEE International Fuzzy Systems Conference*. pp. 1–6 (2007)
17. Astudillo, L., Castillo, O., Melin, P., Alanis, A., Soria, J., Aguilar, L.T.: Intelligent control of an autonomous mobile robot using type-2 fuzzy logic. *Eng. Lett.* **13**, 3 (2006)
18. Wu, D., Tan, W.W.: Genetic learning and performance evaluation of interval type-2 fuzzy logic controllers. *Eng. Appl. Artif. Intell.* **19**(8), 829–841 (2006)
19. Wu, D.: On the fundamental differences between interval type-2 and type-1 fuzzy logic controllers. *IEEE Trans. Fuzzy Syst.* **20**(5), 832–848 (2012)
20. Wu, D., Tan, W.W.: A type-2 fuzzy logic controller for the liquid-level process. In: *2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No. 04CH37542)*, 2, pp. 953–958
21. Yang, X.-S.: *Nature-inspired metaheuristic algorithms*. Luniver Press, Beckington (2010)
22. Yang, X.-S.: Firefly algorithm, lévy flights and global optimization. In: *Research and Development in Intelligent Systems XXVI*. Springer, London, UK, pp. 209–218 (2010)
23. Muthiah-Nakarajan, V., Noel, M.M.: Galactic swarm optimization: a new global optimization metaheuristic inspired by galactic motion. *Appl. Soft Comput.* **38**, 771–787 (2016)
24. Bernal, E., Castillo, O., Soria, J., Valdez, F., Melin, P.: A variant to the dynamic adaptation of parameters in galactic swarm optimization using a fuzzy logic augmentation. In: *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 1–7 (2018)
25. Bernal, E., Castillo, O., Soria, J., Valdez, F.: Galactic swarm optimization with adaptation of parameters using fuzzy logic for the optimization of mathematical functions, pp. 131–140. Springer, Cham (2018)
26. Olivás, F., Valdez, F., Melin, P., Sombra, A., Castillo, O.: Interval type-2 fuzzy logic for dynamic parameter adaptation in a modified gravitational search algorithm. *Inf. Sci.* **476**, 159–175 (2019)
27. Valdez, F., Melin, P., Castillo, O.: An improved evolutionary method with fuzzy logic for combining particle swarm optimization and genetic algorithms. *Appl. Soft Comput.* **11**(2), 2625–2632 (2011)
28. Castillo, O., Neyoy, H., Soria, J., García, M., Valdez, F.: Dynamic fuzzy logic parameter tuning for ACO and its application in the fuzzy logic control of an autonomous mobile robot. *Int. J. Adv. Robot. Syst.* **10**(1), 51 (2013)
29. Amador-Angulo, L., et al.: Fuzzy sets in dynamic adaptation of parameters of a bee colony optimization for controlling the trajectory of an autonomous mobile robot. *Sensors* **16**(9), 1458 (2016)
30. Hossain, M.A., Ferdous, I.: Autonomous robot path planning in dynamic environment using a new optimization technique inspired by bacterial foraging technique. *Rob. Auton. Syst.* **64**, 137–141 (2015)
31. Lee, T.-C., Song, K.-T., Lee, C.-H., Teng, C.-C.: Tracking control of unicycle-modeled mobile robots using a saturation feedback controller. *IEEE Trans. Control Syst. Technol.* **9**(2), 305–318 (2001)
32. Martínez, R., Aguilar, L.T.: Optimization of interval type-2 fuzzy logic controllers for a perturbed autonomous wheeled mobile robot using genetic algorithms. *Inf. Sci.* **179**(13), 2158–2174 (2009)
33. Kanayama, Y., Kimura, Y., Miyazaki, F., Noguchi, T.: A stable tracking control method for an autonomous mobile robot. In: *Proceedings. 1990, IEEE International Conference on Robotics and Automation*, IEEE Press, pp. 384–389
34. Caraveo, C., Valdez, F., Castillo, O.: A new meta-heuristics of optimization with dynamic adaptation of parameters using type-2 fuzzy logic for trajectory control of a mobile robot. *Algorithms* **10**(3), 85 (2017)
35. Peraza, C., Valdez, F., Castro, J.R., Castillo, O.: Fuzzy dynamic parameter adaptation in the harmony search algorithm for the optimization of the ball and beam controller. *Adv. Oper. Res.* (2018). <https://doi.org/10.1155/2018/3092872>
36. Lagunes, M.L., Castillo, O., Soria, J., Garcia, M., Valdez, F.: Optimization of granulation for fuzzy controllers of autonomous mobile robots using the Firefly Algorithm. *Granul. Comput.* **4**(2), 185–195 (2019)
37. Lagunes, M.L., et al.: Multi-metaheuristic competitive model for optimization of fuzzy controllers. *Algorithms* **12**(5), 90 (2019)
38. Eftimov, T., Korošec, P., Seljak, B.K.: A novel approach to statistical comparison of meta-heuristic stochastic optimization algorithms using deep statistics. *Inf. Sci.* **417**, 186–215 (2017)
39. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power. *Inf. Sci.* **180**(10), 2044–2064 (2010)
40. Derrac, J., García, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol. Comput.* **1**(1), 3–18 (2011)

41. Olivas, F., et al.: Comparative study of type-2 fuzzy particle swarm, bee colony and bat algorithms in optimization of fuzzy controllers. *Algorithms* **10**(3), 101 (2017)
42. Ontiveros-Robles, E., Melin, P., Castillo, O.: Comparative analysis of noise robustness of type 2 fuzzy logic controllers. *Kybernetika* **54**, 175–201 (2018)
43. Ontiveros-Robles, E., Melin, P., Castillo, O.: New methodology to approximate type-reduction based on a continuous root-finding Karnik Mendel algorithm. *Algorithms* **10**(3), 77 (2017)



**Emer Bernal** obtained a Ph.D. in computer science from the Tijuana Institute of technology in 2020, in addition to a Master's degree in computer science from the aforementioned institution. His research interests include the type-1 and type-2 fuzzy logic, metaheuristic algorithms, intelligent computing and their applications. He has published more than 15 paper in journals and book chapters.



**Marylu L. Lagunes** received a Master's Degree in Computer Science from the Tijuana Institute of Technology, Tijuana, Mexico, in 2017. She is currently in the process of obtaining a PhD in Computer Science from the Tijuana Institute of Technology. Her research interests include interval type-2 fuzzy logic, dynamic parameter tuning for fuzzy controller membership functions. She has published more than 10 papers in journals and book chapters.



**Oscar Castillo** holds the Doctor in Science degree (Doctor Habilitatus) in Computer Science from the Polish Academy of Sciences (with the Dissertation "Soft Computing and Fractal Theory for Intelligent Manufacturing"). He is a Professor of Computer Science in the Graduate Division, Tijuana Institute of Technology, Tijuana, Mexico. In addition, he is serving as Research Director of Computer Science and head of the research group on Hybrid

Fuzzy Intelligent Systems. Currently, he is President of HAFSA (Hispanic American Fuzzy Systems Association) and Past President of IFSA (International Fuzzy Systems Association). Prof. Castillo is also Chair of the Mexican Chapter of the Computational Intelligence Society (IEEE). He also belongs to the Technical Committee on Fuzzy Systems of IEEE and to the Task Force on "Extensions to

Type-1 Fuzzy Systems". He is also a member of NAFIPS, IFSA and IEEE. He belongs to the Mexican Research System (SNI Level 3). His research interests are in Type-2 Fuzzy Logic, Fuzzy Control, Neuro-Fuzzy and Genetic-Fuzzy hybrid approaches. He has published over 300 journal papers, 10 authored books, 40 edited books, 200 papers in conference proceedings, and more than 300 chapters in edited books, in total more than 800 publications. He has been Guest Editor of several successful Special Issues in the past, like in the following journals: Applied Soft Computing, Intelligent Systems, Information Sciences, Soft Computing, Non-Linear Studies, Fuzzy Sets and Systems, JAMRIS and Engineering Letters. He is currently Associate Editor of the Information Sciences Journal, Applied Soft Computing Journal, Journal of Engineering Applications on Artificial Intelligence, Granular Computing Journal and the IEEE Transactions on Fuzzy Systems. He has been elected IFSA Fellow and MICAI Fellow member last year. Finally, he recently received the Recognition as Highly Cited Researcher in 2017 and 2018 by Clarivate Analytics and Web of Science.



**José Soria** has worked in several fields of Applied Sciences such as Physics from a rigorous mathematical point of view in the fields of Statistical Mechanics. He has also worked in Computer Aided Molecular Modeling and in recent years he has worked in the research group of Dr. Oscar Castillo on Type-2 Fuzzy Logic and Bio-Inspired Optimization. He has to date publications in Statistical Mechanics, String Theory and fuzzy logic. The total number of

publications is more than 100, including journal papers and book chapters. He has directed and co-directed postgraduate thesis in Master and Doctoral programs in Chemistry and Computer Science. Jose Soria has a PhD in Mathematics from the renowned "Courant Institute of Mathematical Sciences in New York University" (NYU) in 1988. Masters in Mathematics at NYU. Master and Bachelor Degree in Physics from the Autonomous University of Mexico (UNAM).



**Fevrier Valdez** holds the Ph. D. in Computer Science from UABC University in Tijuana, Mexico (2009) and the Master's Degree in Computer Science from Tijuana Institute of Technology (2004). Currently, he is Professor of Computer Science in the Division of Graduate Studies and Research of Tijuana Institute of Technology, where he lectures at the under-graduate and graduate levels. His research areas interests are in evolutionary computing, fuzzy

logic, hybrid genetic-fuzzy systems, and new techniques for pattern recognition. He has published over 200 papers (Journals, chapters of books and conference papers). He was Lead Guest Editor of the successful Special Issue Nature-Inspired Optimization Algorithms for Neuro-Fuzzy Models in Real World Control and Robotics Applications of the Journal *Computational Intelligence and Neuroscience*.