

Article

Optimization of Fuzzy Controller Using Galactic Swarm Optimization with Type-2 Fuzzy Dynamic Parameter Adjustment

Emer Bernal, Oscar Castillo *^{ID}, José Soria and Fevrier Valdez^{ID}

Division of Graduate Studies and Research, Tijuana Institute of Technology, 22414 Tijuana, Mexico; emer_recargado@hotmail.com (E.B.); jsoria57@gmail.com (J.S.); fevrier@tectijuana.mx (F.V.)

* Correspondence: ocastillo@tectijuana.mx

Received: 10 February 2019; Accepted: 20 February 2019; Published: 25 February 2019



Abstract: Galactic swarm optimization (GSO) is a recently created metaheuristic which is inspired by the motion of galaxies and stars in the universe. This algorithm gives us the possibility of finding the global optimum with greater precision since it uses multiple exploration and exploitation cycles. In this paper we present a modification to galactic swarm optimization using type-1 (T1) and interval type-2 (IT2) fuzzy systems for the dynamic adjustment of the c_3 and c_4 parameters in the algorithm. In addition, the modification is used for the optimization of the fuzzy controller of an autonomous mobile robot. First, the galactic swarm optimization is tested for fuzzy controller optimization. Second, the GSO algorithm with the dynamic adjustment of parameters using T1 fuzzy systems is used for the optimization of the fuzzy controller of an autonomous mobile robot. Finally, the GSO algorithm with the dynamic adjustment of parameters using the IT2 fuzzy systems is applied to the optimization of the fuzzy controller. In the proposed approaches, perturbation (noise) was added to the plant in order to find out if our approach behaves well under perturbation to the autonomous mobile robot plant; additionally, we consider our ability to compare the results obtained with the approaches when no perturbation is considered.

Keywords: galactic swarm optimization; fuzzy systems; fuzzy controller optimization

1. Introduction

The main goal of the optimization methods is to find the best solution of a set of possible solutions to a problem; for some cases, the search space is very extensive, which leads to a computational cost that is too high. To solve this problem there are different computational intelligence techniques that provide tools to solve these optimization problems [1].

Nowadays, control is one of the most important applications of fuzzy logic because it provides a greater robustness in the design of fuzzy controllers and a higher performance than classically used control tools [2].

Due to this, fuzzy controllers are used to deal with complex plants; however, the advantages offered by fuzzy logic and fuzzy sets, along with the appearance of the type-2 fuzzy logic, amplifies the probability of designing better fuzzy controllers [3,4].

Fuzzy control helps us combine expert knowledge with measurements and processes based on mathematical models. Fuzzy systems convert a knowledge base into a mathematical model that has been shown to be effective in many applications in industry and real life [5,6].

Fuzzy logic can adapt to the real world, it can work and it can understand everyday expressions like “it’s very cold”, “it’s very low”, etc. The key lies in its adaptation to natural language, each fuzzy set having a membership function associated with each element that represents the fact that both elements are part of the set [7].

A fuzzy set is an extension of the traditional sets; its main difference is that in the traditional one an element may or may not belong to that set, while in the fuzzy sets each element may belong to more than one set with a certain membership degree [8].

Fuzzy logic tries to imitate human thought in the computer, whereas classical computation can only handle dual values like false or true, or yes or no. Fuzzy logic uses mathematical models to represent subjective notions such as “big, small” or “hot, cold” for values that can be manipulated by the computer [9].

In recent years, there has been an increase in research on type-1 and type-2 fuzzy logic; published works can be found on parameters adjustment in metaheuristic algorithms, and control that is based on fuzzy logic has been investigated in order to improve performance, as is shown in the following works:

In an improved evolutionary method with fuzzy logic for combining particle swarm optimization and genetic algorithms [10], a hybrid approach is described for the optimization of particle swarm optimization (PSO) and ant colony optimization (ACO) using fuzzy logic for the dynamic adjustment of the parameters of both algorithms.

In the gravitational search algorithm-based design of fuzzy control systems with a reduced parametric sensitivity [11], the design of the fuzzy control systems is proposed using the Gravitational Search Algorithm; the Takagi-Sugeno fuzzy controller, characterized by linear models of the second order with an integral component, are obtained.

Also in model-free sliding mode fuzzy controllers for reverse osmosis desalination plants [12]. In this case, Takagi-Sugeno fuzzy controllers are proposed for flow control and conductivity in desalination plants. The controllers are designed in view of the inequalities of the linear matrix as constraints using the gray wolf algorithm for their optimization.

Muthiah-Nakarajan and Noel proposed the galactic swarm optimization (GSO) as an algorithm that has proven to have a good performance dealing with multimodal problems or with a high number of dimensions; because it presents multiple exploration and exploitation cycles, it offers us the possibility of finding the best solutions without getting stuck [13,14].

This paper presents a modification to the GSO metaheuristic with the use of type-1 fuzzy systems for the adjustment of the c_3 and c_4 parameters in the GSO algorithm [15]. In addition to a type-1 fuzzy system extension, we are also using type-2 fuzzy logic to give rise to an interval type-2 fuzzy system (IT2), and this proposal was used for the optimization of the membership functions of the controller of the autonomous mobile robot [16]. Experiments were carried out optimizing the fuzzy controller for the aforementioned study case, and the optimization was initially carried out with the original galactic swarm optimization, after which the T1 and IT2 fuzzy systems were used to perform the fuzzy controller optimization, in the same way that the original GSO algorithm was experimented with by adding noise to the plant and without noise in order to compare the proposed approach.

The organization of this paper is as follows: Section 2 explains the operation and equations present in the original GSO algorithm. In Section 3 the development of the proposed approach for the fuzzy controller optimization of the autonomous mobile robot is presented. In Section 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 and the autonomous mobile robot plant, and finally in Section 6 the conclusions are presented.

2. Galactic Swarm Optimization

The original galactic swarm optimization algorithm mimics the motion of stars and galaxies in the universe. The stars are not evenly distributed in the cosmos, and they are concentrated in galaxies that in turn are not evenly distributed. The attraction of the stars and galaxies in the GSO algorithm is emulated in the following way [13,15].

First, the initial population is divided into subpopulations which will be called sub-swarms; all the individuals of the sub-swarms begin their movement according to the PSO algorithm by a

determined number of iterations, and all the individuals in each subpopulation will be attracted towards the individual with better fitness; at the end of the iterations each of the subpopulations will be represented by the best individual of each of the subpopulations.

The best individuals of all the subpopulations will pass into a second phase, where they will form a new super swarm, and (in the same way they will move according to the PSO algorithm at the end of the iterations) the GSO algorithm will return us the best individual of the super swarm which will represent the best solution found in the entire initial population.

All the necessary steps to perform the operation of the GSO algorithm are summarized in the following pseudocode, where the two necessary phases of the GSO algorithm are presented [13]:

1. Start GSO.
2. The population is divided into M subpopulations.

$$X_i \subset X : i = 1, 2, \dots, M \tag{1}$$

3. The population is initialized randomly.

$$X_j^{(i)} \in X_i : j = 1, 2, \dots, N \tag{2}$$

4. Begin Level 1.
Begin PSO for each of the M subpopulations, and calculate the position and velocity of the particles.

$$v_j^{(i)} \leftarrow W_1 v^{(i)} + c_1 r_1 (p_j^{(i)} - x_j^{(i)}) + c_2 r_2 (g^{(i)} - x_j^{(i)}) \tag{3}$$

$$x_j^{(i)} \leftarrow x_j^{(i)} + v_j^{(i)} \tag{4}$$

End PSO.

End Level 1.

5. Begin Level 2.
Initialization of the super swarm.

$$Y^{(i)} \in Y : i = 1, 2, \dots, M \tag{5}$$

Begin PSO.

Calculate the position and velocity of the particles.

$$v^{(i)} \leftarrow W_2 v^{(i)} + c_3 r_3 (p^{(i)} - Y^{(i)}) + c_4 r_4 (g - Y^{(i)}) \tag{6}$$

$$Y^{(i)} \leftarrow Y^{(i)} + v^{(i)} \tag{7}$$

End PSO.

End Level 2.

6. Return the best position g and fitness value $f(g)$.
7. End GSO.

Given that the PSO algorithm is used as a basis in galactic swarm optimization, the algorithm consists of exploration and exploitation phases for which it uses parameters in the calculation of the velocity that reduces or increases as time passes to control the exploration phase and move on to the exploitation phase.

By linearly decreasing, the PSO algorithm tends to have a global search capability when starting the execution and a local search when approaching the end of the execution.

These parameters are the constants of acceleration that make the solutions approach the best local and global solutions, and for this they initially take fixed values and change their values on a mathematical form.

Dynamic optimization problems, where the environment changes dynamically, require that the search algorithm has the ability to search non-linearly to follow the changing environment. Therefore, it must be changed dynamically for a better balance between the local and global search.

3. Proposed Approach (FGSO)

This paper presents a modification to the GSO metaheuristic with the use of type-1 (T1) and interval type-2 (IT2) fuzzy logic for the parameter adjustment of the GSO algorithm.

The use of parameter adjustment in metaheuristic algorithms has become a widely used strategy in recent years. The results published by some authors show significant improvements when the dynamic adjustment of parameters in metaheuristic algorithms is used. The use of fuzzy logic in metaheuristic algorithms is a technique that allows algorithms to perform an automatic search for the optimal values of the parameters that play an important role in the performance of metaheuristic algorithms.

The use of non-linear plants for fuzzy control can turn out to be a complex task because the exact knowledge of each of the components or characteristics that make up the plant is very rarely obtained, so fuzzy sets and fuzzy logic principles are used to model the operation of the plant.

The main contribution of this document is the use of T1 and IT2 fuzzy systems for the dynamic adjustment of the c_3 and c_4 parameters as the iterations pass, which will be used for the optimization of the membership functions of the controller of the autonomous mobile robot. The proposed method can be found in Figure 1.

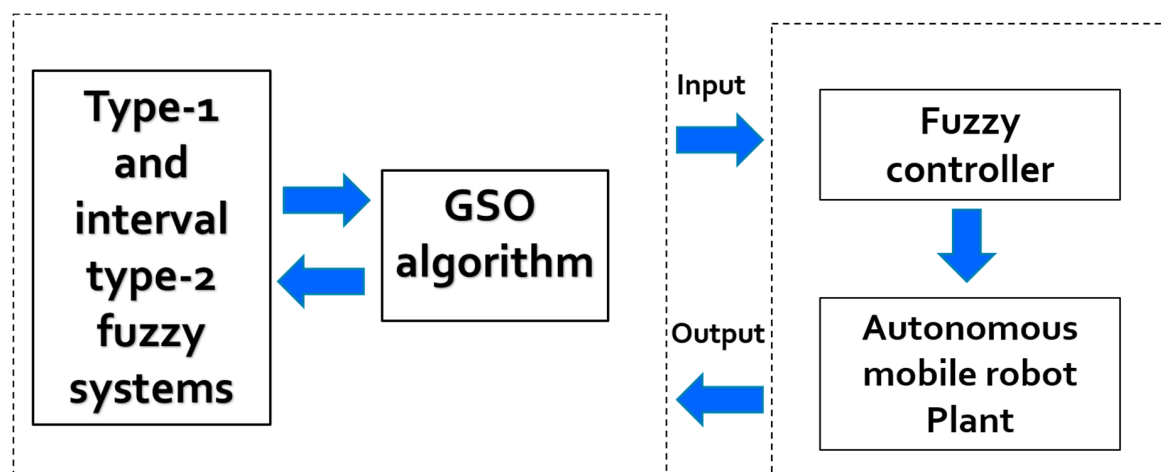


Figure 1. Proposed method.

The optimization of the fuzzy controller of the autonomous mobile robot (AMR) consists in finding the appropriate parameters for the membership functions of the fuzzy controller. Here, each individual used in our proposal will contain the necessary parameters to generate a fuzzy controller, which will be tested in the hopes of obtaining better results than those that can be obtained so far by establishing the parameters of the membership functions based on experience. In Figure 2 we can find the composition of the individuals used in our proposed methodology [10,17,18].

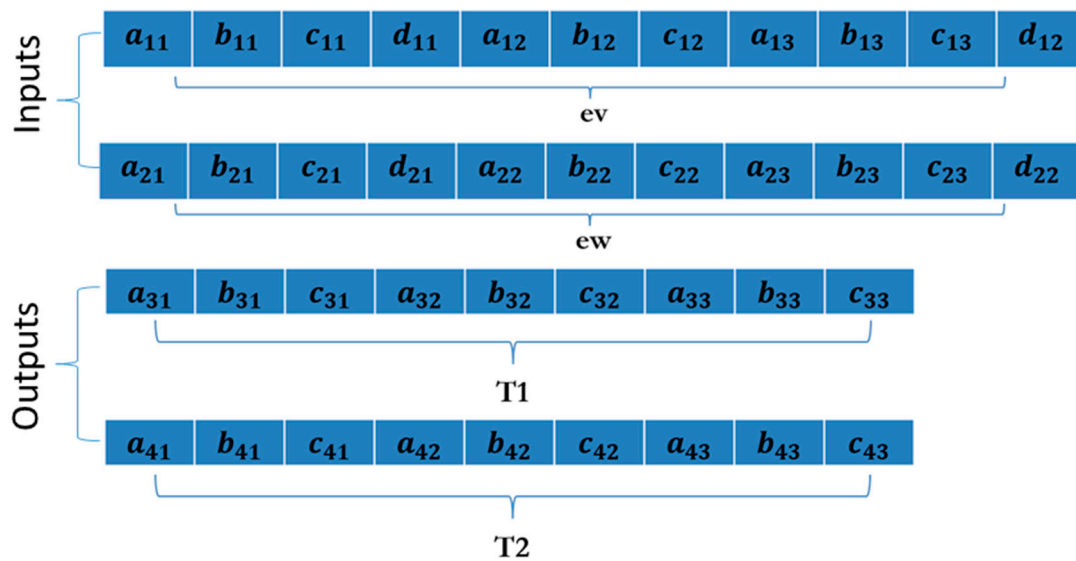


Figure 2. Composition of the individuals in GSO and the proposal fuzzy galactic swarm optimization type-1 (FGSO1) and fuzzy galactic swarm optimization interval type-2 (FGSO1T2).

The type-1 fuzzy set is characterized by having a membership function that can take values between 0 and 1, and is represented as a set of ordered pairs of a certain element; its degree of belonging to said set is defined by the following expression [19,20]:

$$A = \{(x, \mu_A(x)) | x \in X\} \tag{8}$$

where $\mu_A(x)$ is the membership degree of each element.

T1 fuzzy logic was proposed as an imprecision of human language, where the degree of belonging in a range of 0 to 1 is considered.

The appearance of IT2 fuzzy logic provides a more complete model that considers the vagueness and uncertainty, which increases the robustness and improves the performance of your applications [21–23].

An IT2 fuzzy set is represented by two membership functions, a lower and an upper; the area that lies between these membership functions is called a footprint of uncertainty. The mathematical expression is as follows [24,25]:

$$\tilde{A} = \{((x, u), 1) | \forall x \in X, \forall u \in J_x \subseteq [0, 1]\} \tag{9}$$

where $\mu_{\tilde{A}}(x, u)$ represents the membership degree of each element, X is the primary domain and J_x is the secondary domain.

The design of the IT2 fuzzy system is based on a T1 fuzzy system previously used [15], these fuzzy systems have an iteration as an input variable that is formed by three triangular membership functions; the c_3 and c_4 parameters represent the output variables that likewise have 3 triangular membership functions. The design of these fuzzy systems is shown in Figures 3 and 4.

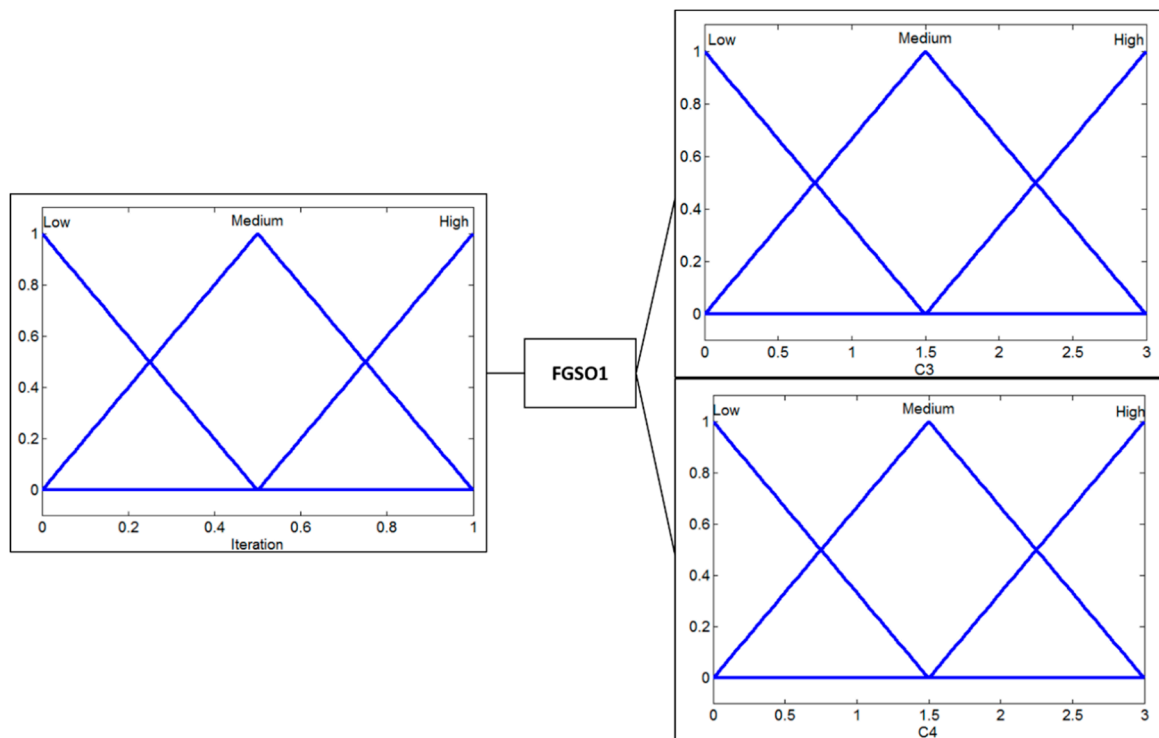


Figure 3. T1 fuzzy system FGS01.

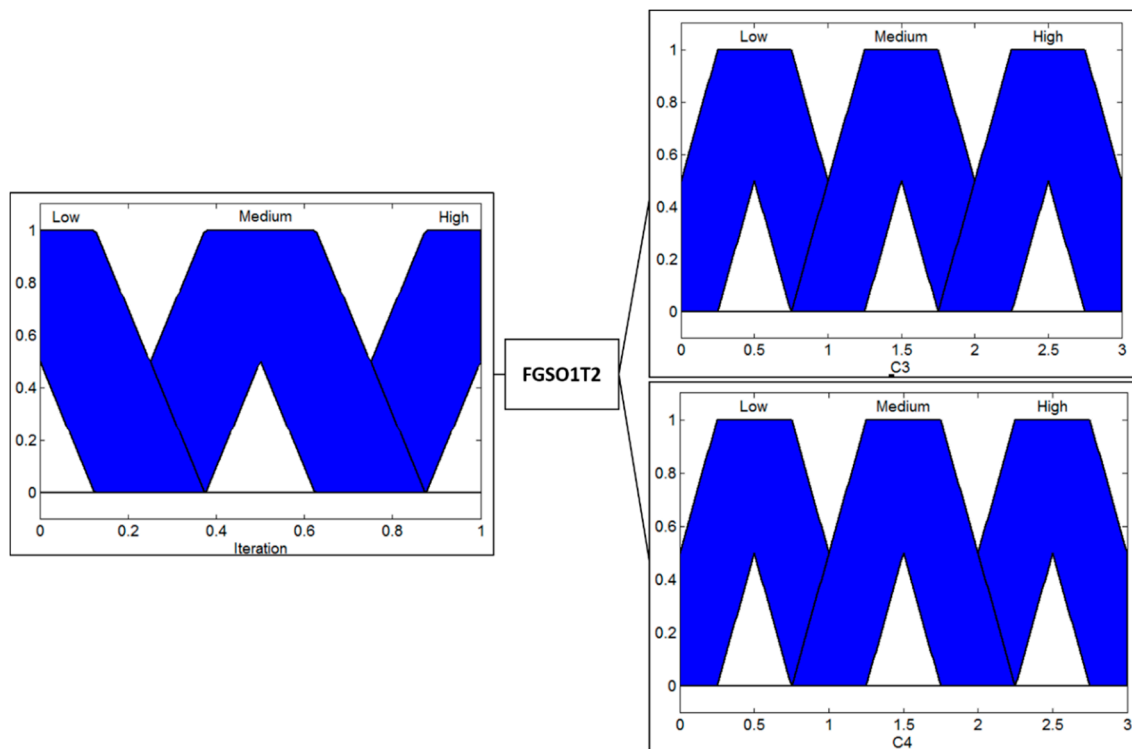


Figure 4. IT2 fuzzy system FGS01T2.

The variable iteration is obtained by means of a percentage of the current iterations and the total of the iterations with which the algorithm works; the expression used for the calculation is presented below [26,27]:

$$\text{Iteration} = \frac{\text{Current Iteration}}{\text{Total Iterations}} \tag{10}$$

The representation of the triangular membership functions of the T1 fuzzy system consists of 3 parameters a, b and c , where $a < b < c$ is shown as follows [20]:

$$\text{triangle}(x; a, b, c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (11)$$

where a and c are the parameters that represent the edges of the triangle, and the parameter b is the center for forming the triangular membership function, as shown in Figure 5.

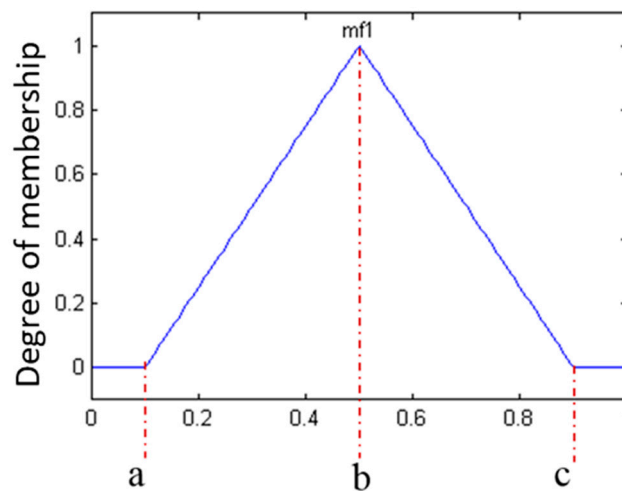


Figure 5. T1 triangular membership function.

In the design of the T1 fuzzy system, input and output variables are used: in this case, one input and two outputs. The variables as mentioned above are the iterations representing the input variables, and the c_3 and c_4 parameters are the iterations representing the output variables. The knowledge representation of the variables is shown in the following equations [28]:

$$\begin{aligned} \mu_{Low}(x) &= \begin{cases} 0, & x \leq -0.5 \\ \frac{x+0.5}{0+0.5}, & -0.5 \leq x \leq 0 \\ \frac{0.5-x}{0.5-0}, & 0 \leq x \leq 0.5 \\ 0, & 0.5 \leq x \end{cases} \\ \mu_{Medium}(x) &= \begin{cases} 0, & x \leq 0 \\ \frac{x-0}{0.5-0}, & 0 \leq x \leq 0.5 \\ \frac{1-x}{1-0.5}, & 0.5 \leq x \leq 1 \\ 0, & 1 \leq x \end{cases} \\ \mu_{High}(x) &= \begin{cases} 0, & x \leq 0.5 \\ \frac{x-0.5}{1-0.5}, & 0.5 \leq x \leq 1 \\ \frac{1.5-x}{1.5-1}, & 1 \leq x \leq 1.5 \\ 0, & 1.5 \leq x \end{cases} \end{aligned} \quad (12)$$

The representation of the triangular membership functions of the IT2 fuzzy system consists of 6 parameters a_1, b_1, c_1, a_2, b_2 and c_2 , where $a_1 < a_2, b_1 < b_2$ and $c_1 < c_2$ is shown as follows [29]:

$$\mu(x) = \text{itritype2}(x, [a_1, b_1, c_1, a_2, b_2, c_2])$$

$$\mu_1(x) = \begin{cases} 0, & x \leq a_1 \\ \frac{x - a_1}{b_1 - a_1} & a_1 \leq x \leq b_1 \\ \frac{c_1 - x}{c_1 - b_1} & b_1 \leq x \leq c_1 \\ 0, & c_1 \leq x \end{cases} \tag{13}$$

$$\mu_2(x) = \begin{cases} 0, & x \leq a_2 \\ \frac{x - a_2}{b_2 - a_2} & a_2 \leq x \leq b_2 \\ \frac{c_2 - x}{c_2 - b_2} & b_2 \leq x \leq c_2 \\ 0, & c_2 \leq x \end{cases}$$

where a_1, a_2, c_1 and c_2 are the parameters that represent the edges of the triangle, and the parameters b_1 and b_2 are the center for forming the IT2 triangular membership function, as shown in Figure 6 [30,31].

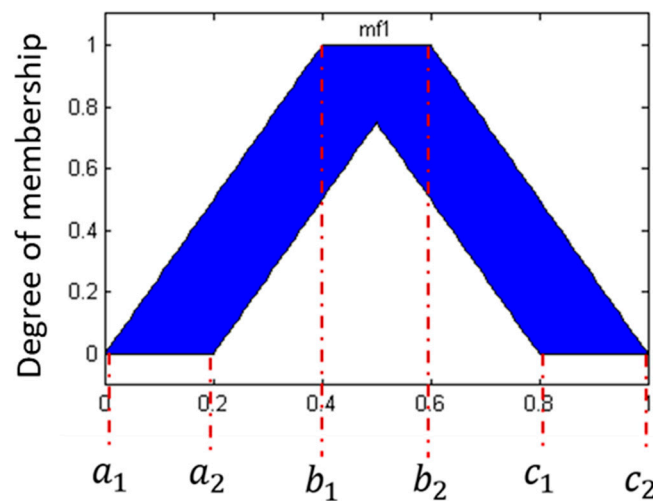


Figure 6. IT2 triangular membership function.

For the design of the IT2 fuzzy system, the T1 fuzzy system presented above is used as a base, where the iteration represents the input variable and the c_3 and c_4 parameters the output variables. The knowledge representation of the IT2 fuzzy system variables is presented in the following equations:

$$\begin{aligned}
 \text{Low } \mu_1(x) &= \begin{cases} 0, & x \leq -0.375 \\ \frac{x + 0.375}{-0.125 + 0.375} & -0.375 \leq x \leq -0.125 \\ \frac{0.125 - x}{0.125 + 0.125} & -0.125 \leq x \leq 0.125 \\ 0, & 0.125 \leq x \end{cases} \\
 \text{Low } \mu_2(x) &= \begin{cases} 0, & x \leq -0.125 \\ \frac{x + 0.125}{0.125 + 0.125} & -0.125 \leq x \leq 0.125 \\ \frac{0.375 - x}{0.375 - 0.125} & 0.125 \leq x \leq 0.375 \\ 0, & 0.375 \leq x \end{cases} \\
 \text{Medium } \mu_1(x) &= \begin{cases} 0, & x \leq 0.125 \\ \frac{x - 0.125}{0.375 - 0.125} & 0.125 \leq x \leq 0.375 \\ \frac{0.625 - x}{0.625 - 0.375} & 0.375 \leq x \leq 0.625 \\ 0, & 0.625 \leq x \end{cases} \\
 \text{Medium } \mu_2(x) &= \begin{cases} 0, & x \leq 0.375 \\ \frac{x - 0.375}{0.625 - 0.375} & 0.375 \leq x \leq 0.625 \\ \frac{0.875 - x}{0.875 - 0.625} & 0.625 \leq x \leq 0.875 \\ 0, & 0.875 \leq x \end{cases} \\
 \text{High } \mu_1(x) &= \begin{cases} 0, & x \leq 0.625 \\ \frac{x - 0.625}{0.875 - 0.625} & 0.625 \leq x \leq 0.875 \\ \frac{1.125 - x}{1.125 - 0.875} & 0.875 \leq x \leq 1.125 \\ 0, & 1.125 \leq x \end{cases} \\
 \text{High } \mu_2(x) &= \begin{cases} 0, & x \leq 0.875 \\ \frac{x - 0.875}{1.125 - 0.875} & 0.875 \leq x \leq 1.125 \\ \frac{1.25 - x}{1.25 - 1.125} & 1.125 \leq x \leq 1.25 \\ 0, & 1.25 \leq x \end{cases}
 \end{aligned} \tag{14}$$

The design of the fuzzy rules is the same, given that the IT2 fuzzy system is an extension of the T1 fuzzy system; both are of the Mamdani type, the first using the T1 fuzzy logic and the second using the T2 fuzzy logic. These fuzzy systems have 3 fuzzy rules, where the idea is that, as the algorithm begins to work, they explore; and as the iterations pass, they explore, as shown in Figure 7 [15].

- 1) If (Iteration is Low) then (C3 is High)(C4 is Low)
- 2) If (Iteration is Medium) then (C3 is Medium)(C4 is Medium)
- 3) If (Iteration is High) then (C3 is Low)(C4 is High)

Figure 7. Rules for FGSO1 and FGSO1T2.

4. Autonomous Mobile Robot

To test the proposed methodology with the dynamic adjustment of the parameters using the T1 and IT2 fuzzy systems, we consider the case of an autonomous mobile robot. For this particular case the fuzzy controller design is optimized to control an autonomous mobile robot, and the main function of the fuzzy controller is to provide a control of the motors to minimize the error in a pre-established trajectory [5,32].

The design of the robot is presented in Figure 8, which has 2 wheels with motors and a passive wheel for the stabilization of the same. The operation of the autonomous mobile robot model is determined by the following equations [33–35]:

$$M(q)v + C(q, \dot{q})v + Dv = \tau + P(\tau), \tag{15}$$

$$\dot{q} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \tag{16}$$

where $q = (x, y, \theta)^T$ is the coordinate vector, which describes the position of the robot, $(v, w)^T$ represents the linear and angular velocity vector, $\tau = (\tau_1, \tau_2)$ is the torque vector applied to the wheels of the robot, τ_1 and τ_2 represent the right and left wheels, $P \in R^2$ is the uniform disturbance vector, $M(q) \in R^{2 \times 2}$ is a symmetric and positive inertial matrix, $C(q, \dot{q})v$ is the vector of the centripetal and Coriolis forces, and $D \in R^{2 \times 2}$ is a diagonal positive defined damping matrix [36,37].

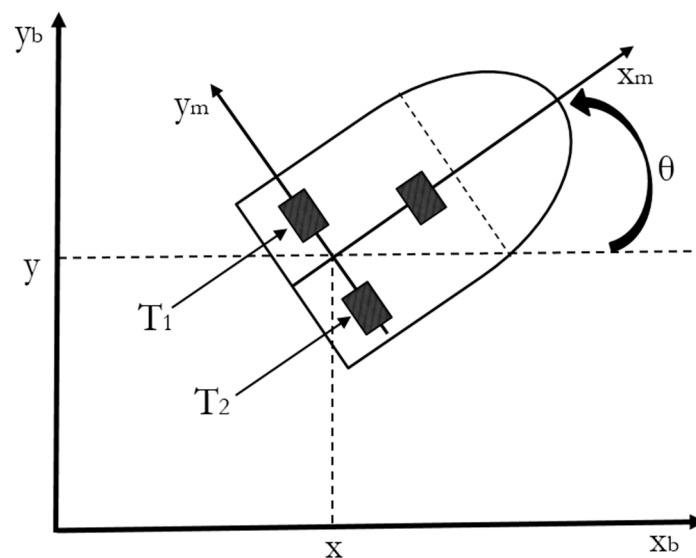


Figure 8. Autonomous mobile robot model.

Fuzzy Controller

The fuzzy system of the mobile robot controller consists of two input variables and two output variables; the first input variable is ev (error in linear velocity), the second input variable is ew (error in angular velocity), and both input variables have two trapezoidal membership functions at the ends and a triangle at the center labeled N (negative), Z (zero) and P (positive) [1,38].

The first output variable is τ_1 (torque 1), the second output variable is τ_2 (torque 2); for the output variables, three triangular membership functions labeled N, Z and P are used. The design of the fuzzy controller of the autonomous mobile robot can be found in Figure 9 [4,33].

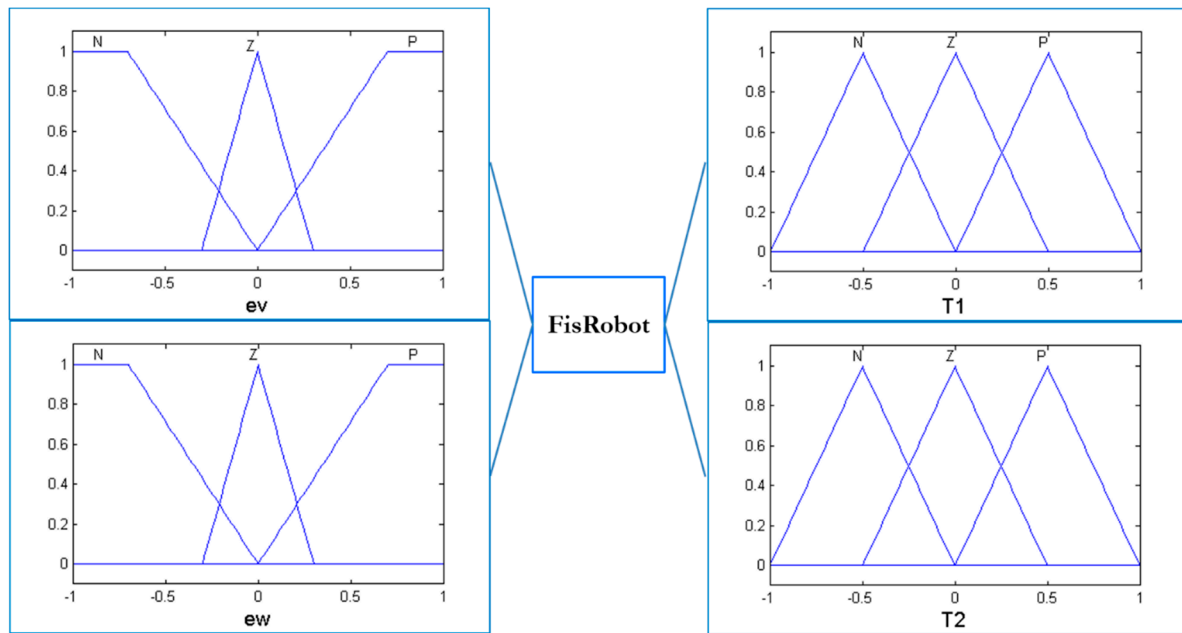


Figure 9. Fuzzy system for the fuzzy controller.

The fuzzy controller rules used to control the dynamics of the plant, in this case the autonomous mobile robot, are shown in Figure 10.

- 1) If (ev is N) and (ew is N) then (T1 is N)(T2 is N)
- 2) If (ev is N) and (ew is Z) then (T1 is N)(T2 is Z)
- 3) If (ev is N) and (ew is P) then (T1 is N)(T2 is P)
- 4) If (ev is Z) and (ew is N) then (T1 is Z)(T2 is N)
- 5) If (ev is Z) and (ew is Z) then (T1 is Z)(T2 is Z)
- 6) If (ev is Z) and (ew is P) then (T1 is Z)(T2 is P)
- 7) If (ev is P) and (ew is N) then (T1 is P)(T2 is N)
- 8) If (ev is P) and (ew is Z) then (T1 is P)(T2 is Z)
- 9) If (ev is P) and (ew is P) then (T1 is P)(T2 is P)

Figure 10. Fuzzy rules for the autonomous mobile robot.

The desired trajectory for the robot to follow begins at the point (0, 0) and turns in order to generate a “U” shaped trajectory; furthermore, to create a more complex problem, the trajectory is repeated on several occasions, as shown in Figure 11.

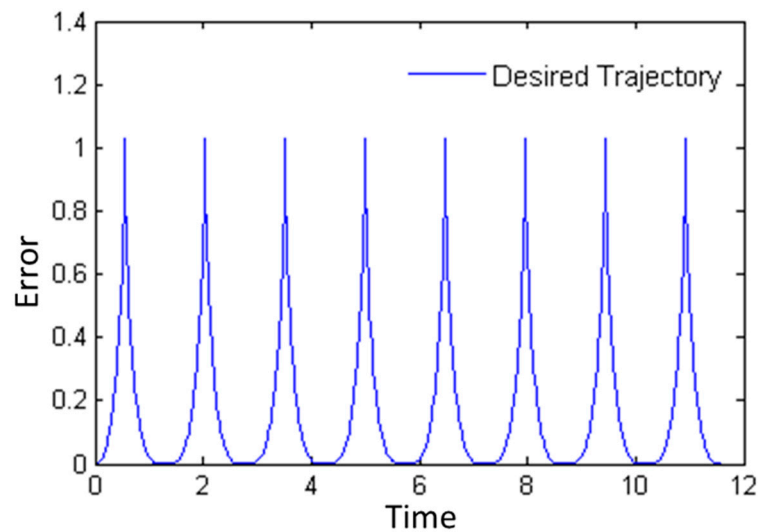


Figure 11. Desired trajectory for the autonomous mobile robot [39].

With the intention of providing better comparison tools between type-1 and interval type-2 fuzzy systems, a modification is made to the autonomous mobile robot, where random number generators are added to simulate the noise in the plant with a level of 5% [40].

After adding noise to the plant, the trajectory of the desired reference varies slightly with respect to the original, as can be seen in Figure 12.

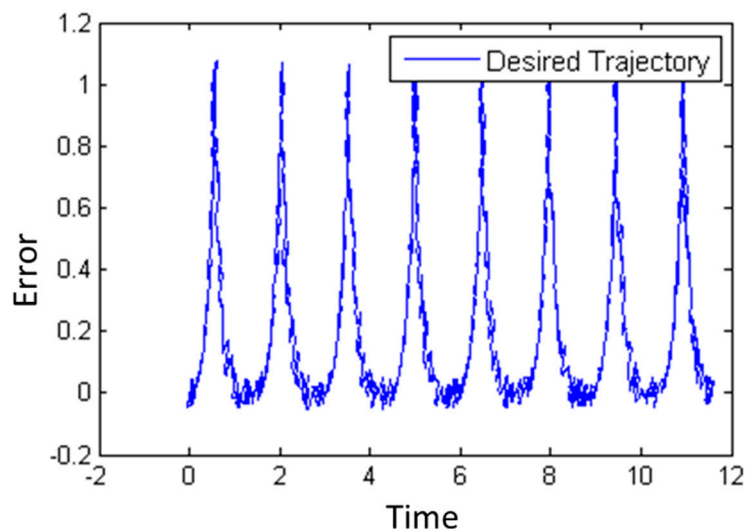


Figure 12. Desired trajectory for the autonomous mobile robot, with noise.

5. Experimental Results

The study case of the autonomous mobile robot is used to test the proposal described in Section 3, where the T1 and IT2 fuzzy systems are used for the dynamic adjustment of the c_3 and c_4 parameters used in the galactic swarm optimization.

The galactic swarm optimization was applied to the optimization of the parameter values of the membership functions of the autonomous mobile robot controller, and the GSO parameters used to perform the optimization are shown in Table 1.

Table 1. Parameters for the optimization in GSO, FGSO1 and FGSO1T2.

Parameter	GSO	FGSO1	FGSO1T2
Population	10	10	10
Subpopulation	5	5	5
Iteration 1	15	15	15
Iteration 2	50	50	50
c ₁ and c ₂	2	Dynamic	Dynamic
c ₃ and c ₄	2	Dynamic	Dynamic

To evaluate the performance of our proposal and the original galactic swarm optimization, a series of performance indices are used, such as the mean square error (MSE), which is described in the below equation [32,41]:

$$MSE = \frac{1}{n} \sum_{i=1}^n (X_i - Y_i)^2 \tag{17}$$

where X_i is the reference value at time i , Y_i is the value produced by the system at time i and N is the number of samples considered.

In order to perform a comparison with the GSO algorithm some variants of the original GSO algorithm were used: the first is the fuzzy galactic swarm optimization (FGSO1) using the T1 fuzzy systems, the second is the FGSO1T2 using the IT2 fuzzy systems. The results obtained with the variants and the original GSO algorithm are presented in Tables 2 and 3.

Table 2. Results without noise for GSO.

Autonomous Mobile Robot			
MSE	GSO	FGSO1	FGSO1T2
Best	4.55×10^{-6}	1.92×10^{-6}	6.48×10^{-4}
Worst	9.67×10^{-2}	9.15×10^{-3}	1.60×10^{-1}
Average	1.14×10^{-2}	1.09×10^{-3}	2.19×10^{-2}
Standard Deviation	2.25×10^{-2}	2.57×10^{-3}	3.56×10^{-2}

Table 3. Results with noise for GSO.

Autonomous Mobile Robot			
MSE	GSO	FGSO1	FGSO1T2
Best	1.45×10^{-1}	2.14×10^{-1}	7.14×10^{-4}
Worst	1.1565	7.84×10^{-1}	9.37×10^{-1}
Average	7.28×10^{-1}	4.70×10^{-1}	4.61×10^{-1}
Standard deviation	2.68×10^{-1}	1.29×10^{-1}	2.22×10^{-1}

In Table 2 we find the average, the best, the worst and the standard deviation of the results obtained after 30 simulations of the mobile autonomous robot plant without adding noise or disturbances. The results show that our proposal FGSO1, using the T1 fuzzy systems, is where better results are obtained on average, followed by the GSO algorithm and finally the proposed FGSO1T2 using the IT2 fuzzy systems. The averages of the results for each of the proposed methods are illustrated in Figure 13.

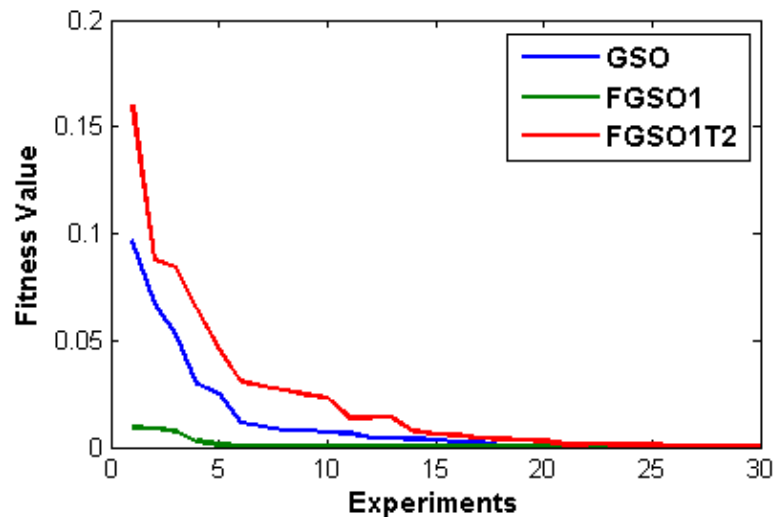


Figure 13. Results for GSO, FGS01 and FGS01T2 without noise.

In Table 3 we find the mean, the best, the worst and the standard deviation of the results obtained after 30 simulations of the mobile autonomous robot plant adding a noise level of 5%. The results show that our proposal FGS01T2 using the IT2 fuzzy systems is where we obtain better results on average, followed by the proposal FGS01 using the T1 fuzzy and finally the GSO algorithm. The averages of the results for each of the proposed methods are illustrated in Figure 14.

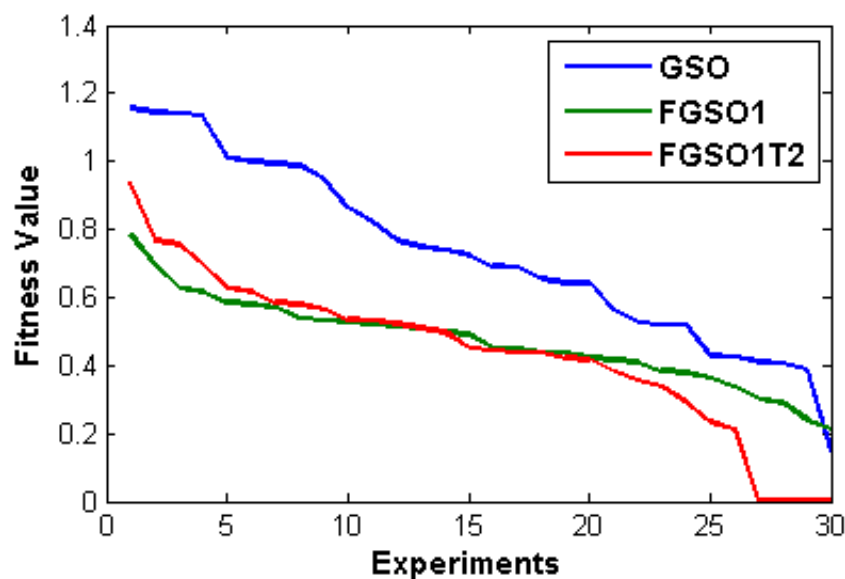


Figure 14. Results for GSO, FGS01 and FGS01T2 with noise.

The figures shown below represent the best trajectories found after performing the simulations of the autonomous mobile robot plant. Figure 15 shows the best simulation obtained with the proposed method FGS01 in the experiments carried out without noise. In Figure 16, we find the best simulation with noise in the autonomous mobile robot plant with the proposed method FGS01T2.

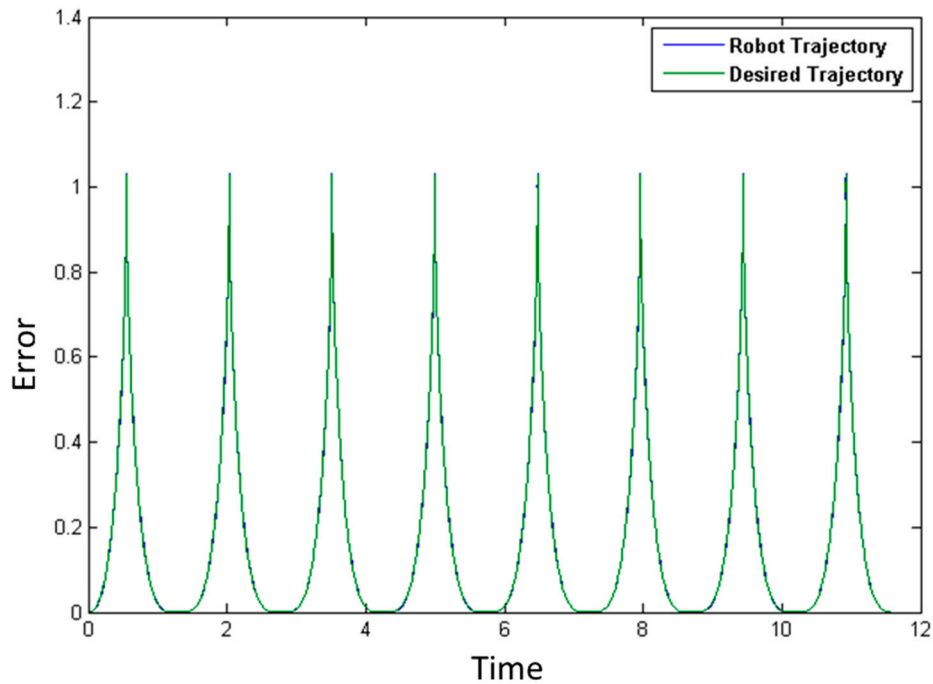


Figure 15. Best trajectory for FGSO1 without noise.

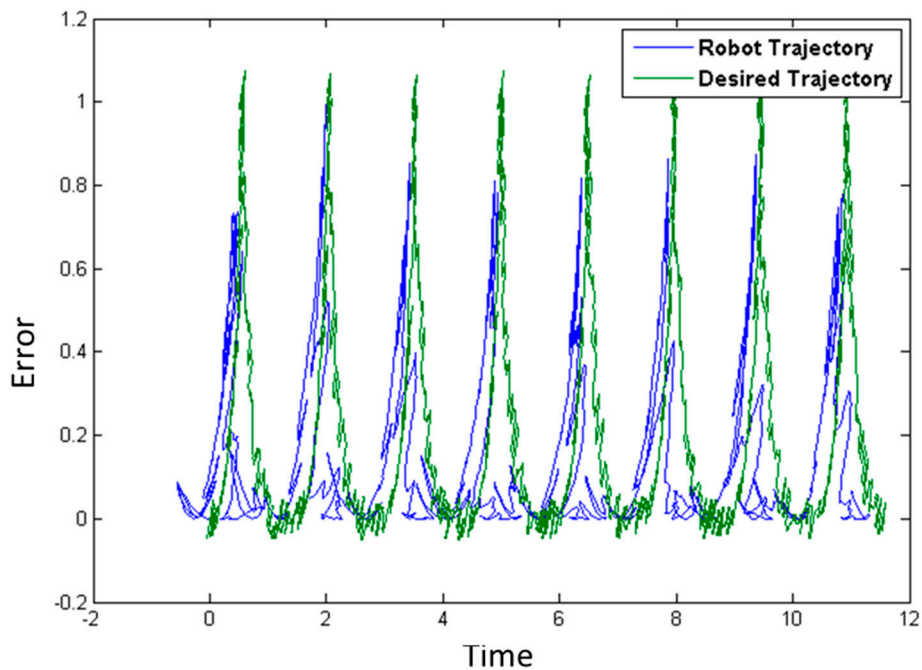


Figure 16. Best trajectory for FGSO1T2 with noise.

The best controllers found with our proposed approach and the original GSO algorithm are shown below, where we can note the arrangement of the points of the membership functions after having performed the optimization of the fuzzy controller with each one of the proposals presented in the previous section. The first controller is shown in Figure 17, obtained by performing the optimization with the original GSO algorithm; the second controller is shown in Figure 18, found using our proposal FGSO1 using the T1 fuzzy systems, and finally the controller shown in Figure 19 is obtained after performing the optimization of the controller with the proposal FGSO1T2 using the IT2 fuzzy systems.

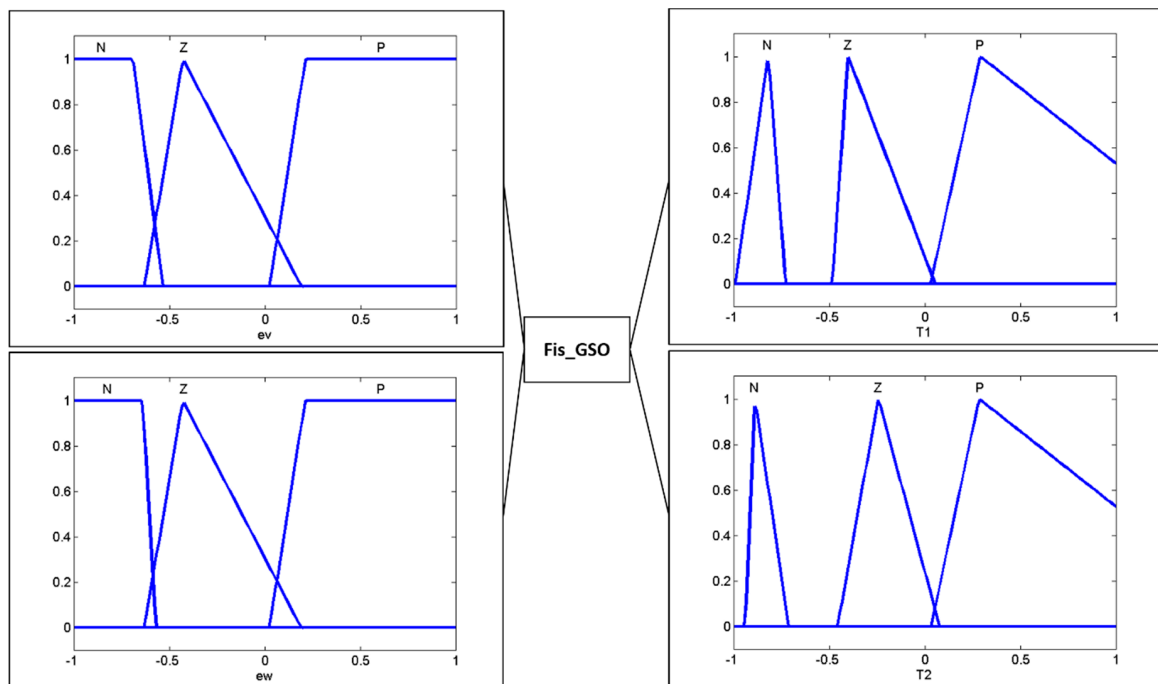


Figure 17. Fuzzy controller optimized with GSO.

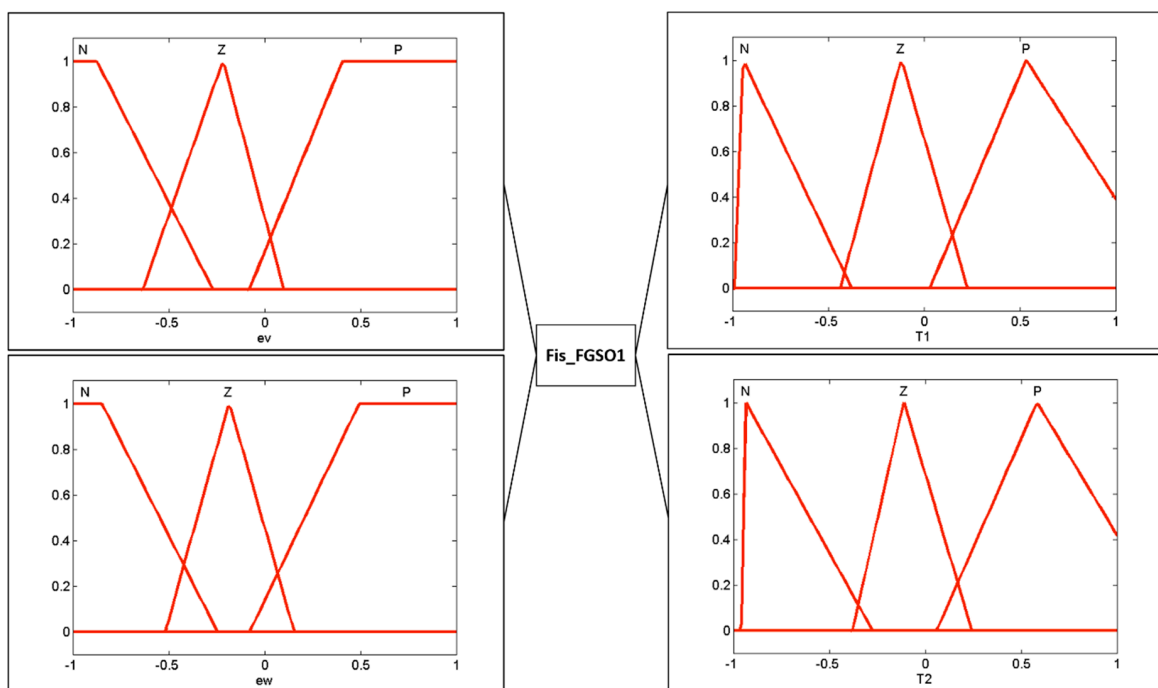


Figure 18. Fuzzy controller optimized with FGS01.

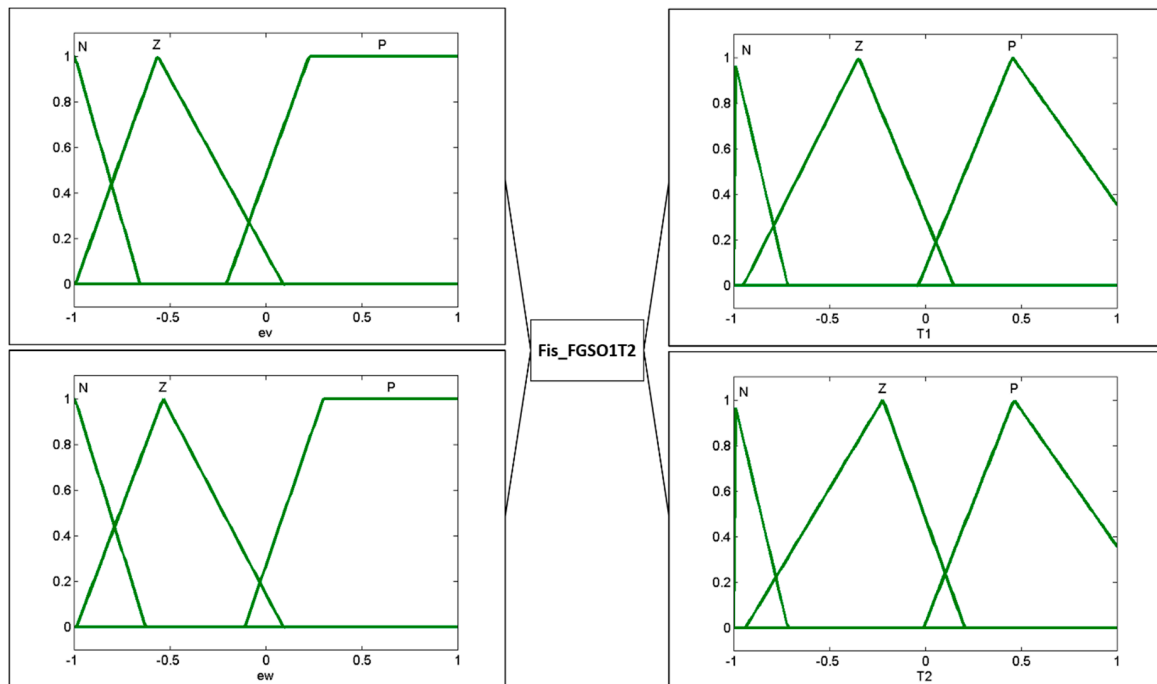


Figure 19. Fuzzy controller optimized with FGSO1T2.

Statistical Comparison

In order to validate the performance of our proposed method using the T1 and IT2 fuzzy systems for the optimization of the membership functions of the autonomous mobile robot, a statistical comparison is made to find evidence that our proposed method yields competitive results for the case study of the autonomous mobile robot plant.

First, the statistical comparison is made between the original GSO and our proposed method that uses the T1 and IT2 fuzzy systems for the adjustment of the c_3 and c_4 parameters. Then a comparison is made with the PSO algorithm. All comparisons are made by experimenting with the autonomous mobile robot plant without adding noise or disturbances to the simulations and adding 5% noise in the simulations of the autonomous mobile robot plant.

The statistical test used to perform the statistical comparison between the methods mentioned above is the z test. In Table 4, we can find the parameters used [17,28].

Table 4. Parameters z test.

Parameter	Value
H_0	$\mu_1 \geq \mu_2$
H_a	$\mu_1 < \mu_2$ (Claim)
Level of Significance	95%
α	0.05
Critical Value	-1.645

H_0 is the null hypothesis; it establishes that the average of the proposals FGSO1 and FGSO1T2 (μ_1) is greater than or equal to the average of the original GSO algorithm (μ_2); furthermore, H_a is the alternative hypothesis (claim) that establishes that the average of the proposals FGSO1 and FGSO1T2 (μ_1) is less than the average of the original GSO algorithm (μ_2).

In Table 5, the results with a 95% level of significance and a rejection zone for values lower than -1.645 are presented. The results of the statistical test for the experiments of the simulations without adding noise to the plant of the autonomous mobile robot give values of -2.4936 for the first comparison, rejecting H_0 and accepting H_a ; there is therefore enough evidence to affirm that the

average of the proposal FGSO1 is lower than the average of the original GSO algorithm; there is a z value of 1.3656 for the second comparison, accepting H_0 , so that the average of the proposal FGSO1T2 is greater than or equal to the average of the original GSO algorithm.

Table 5. Results for the z test with FGSO1, FGSO1T2 and the GSO without noise.

Autonomous Mobile Robot			
Methods	Average	Standard Deviation	z Value
FGSO1 (μ_1)	1.09×10^{-3}	2.57×10^{-3}	−2.4936
GSO (μ_2)	1.14×10^{-2}	2.25×10^{-2}	
FGSO1T2 (μ_1)	2.19×10^{-2}	3.56×10^{-2}	1.3656
GSO (μ_2)	1.14×10^{-2}	2.25×10^{-2}	

Table 6 show the results with a 95% level of significance and a rejection zone for values lower than −1.645. The results of the statistical test for the experiments of the simulations with a 5% noise level in the signal of the plant of the autonomous mobile robot give values of −4.7511 and −4.2023. As a result, the H_0 is rejected and the H_a is accepted; there is significant evidence to affirm that the average of the proposals FGSO1 and FGSO1T2 is lower than the average of the original GSO algorithm.

Table 6. Results for the z test with FGSO1, FGSO1T2 and the GSO with noise.

Autonomous Mobile Robot			
Methods	Average	Standard Deviation	z Value
FGSO1 (μ_1)	0.4700	0.1290	−4.7511
GSO (μ_2)	0.7280	0.2680	
FGSO1T2 (μ_1)	0.4610	0.2220	−4.2023
GSO (μ_2)	0.7280	0.2680	

The H_0 null hypothesis establishes that the average of the original GSO algorithm and the proposals FGSO1 and FGSO1T2 (μ_1) is greater than or equal to the average of the original PSO algorithm, and the variants PSO + T1FS and PSO + IT2FS (μ_2); furthermore, the H_a alternative hypothesis (claim) establishes that the average of the original GSO algorithm and the proposals FGSO1 and FGSO1T2 (μ_1) is less than the average of the original PSO algorithm, and the variants PSO + T1FS and PSO + IT2FS (μ_2).

Table 7 shows the results with a 95% level of significance and a rejection zone for values lower than −1.645. The results of the statistical test for the experiments of the simulations without adding noise to the plant of the autonomous mobile robot give values of −4.4229 and −5.1972, rejecting H_0 and accepting H_a . There is therefore enough evidence to affirm that the average of the original GSO algorithm and the proposal FGSO1 is lower than the average of the original PSO algorithm and the variant PSO + T1FS. Finally, with a z value of −0.7116 for the comparison between the proposal FGSO1T2 and the variant PSO + IT2FS, H_0 is accepted; so the average of the proposal FGSO1T2 is greater than or equal to the average of the variant PSO + IT2FS.

Table 8 shows the results with a 95% level of significance and a rejection zone for values lower than −1.645. The results of the statistical test for the experiments of the simulations with a 5% noise level in the signal of the plant of the autonomous mobile robot give values of −11.3293, −29.3360 and −26.0009, so H_0 is rejected and the H_a is accepted; there is significant evidence to affirm the average of the original.

The GSO algorithm and the proposals FGSO1 and FGSO1T2 are lower than the average of the original PSO algorithm, and of the variants PSO + T1FS and PSO + IT2FS.

Table 7. Results for the z test between the proposal and PSO [4] without noise.

Autonomous Mobile Robot			
Methods	Average	Standard Deviation	z Value
GSO (μ_1)	1.14×10^{-2}	2.25×10^{-2}	−4.4229
PSO (μ_2)	2.4166	2.9784	
FGSO1 (μ_1)	1.09×10^{-3}	2.57×10^{-3}	−5.1972
PSO + T1FS (μ_2)	0.3042	0.3194	
FGSO1T2 (μ_1)	2.19×10^{-2}	3.56×10^{-2}	−0.7116
PSO + IT2FS (μ_2)	3.22×10^{-2}	7.11×10^{-2}	

Table 8. Results for the z test between the proposal and PSO [4] with noise.

Autonomous Mobile Robot			
Methods	Average	Standard Deviation	z Value
GSO (μ_1)	0.7280	0.2680	−11.3293
PSO (μ_2)	6.4302	2.7437	
FGSO1 (μ_1)	0.4700	0.1290	−29.3360
PSO + T1FS (μ_2)	3.1108	0.4759	
FGSO1T2 (μ_1)	0.4610	0.2220	−26.0009
PSO + IT2FS (μ_2)	2.3630	0.3335	

6. Conclusions

The main contribution in this paper is the study of the original galactic swarm optimization: of the variants FGSO1 using type-1 fuzzy systems and the variants FGSO1T2 using type-2 fuzzy systems, applied in the optimization controller of the mobile robot. Type-1 and interval type-2 fuzzy logic show a good management of the uncertainty in complex problems where disturbances or noise are present in the plant, and fuzzy controllers maintain stability in the simulation results.

For the results obtained without using noise or disturbances, the variant using type-1 fuzzy logic FGSO1 was the one with better results, thus obtaining a better controller than the original GSO. With the use of noise in the plant signal, the variants FGSO1 and FGSO1T2 manage to improve the obtained results with respect to the original GSO algorithm. The best overall fuzzy controller design was found with FGSO1T2, even if on average it could not show a statistical advantage.

In order to compare the performance of our proposal against other metaheuristics, a comparison was made with respect to the PSO algorithm. For the simulations carried out without adding noise to the mobile autonomous robot plant; our variants GSO, FGSO1 and FGSO1T2 managed to overcome the results obtained by the PSO algorithm, as well as its variants PSO + T1FS and PSO + IT2FS.

In the comparison made by adding noise to the plant in the same way, our proposal offered more competitive results than those obtained with PSO and its variants, and all this is based on the results presented in Section 5.

We can conclude that the GSO algorithm and the proposed variants fuzzy galactic swarm optimization (FGSO1) and FGSO1T2 using the IT2 fuzzy systems show competitive results when applied to the optimization of the membership functions from the study case of the autonomous robot mobile. As a result, we will continue to investigate this area to even further explore the behavior of the GSO algorithm and the use of the type-1 and type-2 fuzzy logic, since good results have been presented in control problems and in the optimization of the benchmark mathematical functions according to the current study and previous publications.

Author Contributions: E.B. analyzed the original metaheuristic methods and applied Type-1 and Type-2 fuzzy logic for dynamic parameter adaptation applied for tracking trajectories, and contributed to performing the experiments and wrote the paper; O.C. reviewed the state of the art, analyzed the data and proposed the method;

F.V. contributed to the discussion and analysis of the results; J.S. contributed to the formal analysis of the results and the equations of the methods.

Funding: This research was funded by CONACYT under grant 122.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. 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* **2017**, *10*, 85. [[CrossRef](#)]
2. Ontiveros-Robles, E.; Melin, P.; Castillo, O. Comparative analysis of noise robustness of type 2 fuzzy logic controllers. *Kybernetika* **2018**, *54*, 175–201. [[CrossRef](#)]
3. Ontiveros-Robles, E.; Melin, P.; Castillo, O. New Methodology to Approximate Type-Reduction Based on a Continuous Root-Finding Karnik Mendel Algorithm. *Algorithms* **2017**, *10*, 77. [[CrossRef](#)]
4. Olivas, F.; Amador-Angulo, L.; Perez, J.; Caraveo, C.; Valdez, F.; Castillo, O. Comparative Study of Type-2 Fuzzy Particle Swarm, Bee Colony and Bat Algorithms in Optimization of Fuzzy Controllers. *Algorithms* **2017**, *10*, 101. [[CrossRef](#)]
5. 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.* **2013**, *10*, 51. [[CrossRef](#)]
6. Hasanien, H.M.; Matar, M. A Fuzzy Logic Controller for Autonomous Operation of a Voltage Source Converter-Based Distributed Generation System. *IEEE Trans. Smart Grid* **2015**, *6*, 158–165. [[CrossRef](#)]
7. Chen, G.; Tat Pham, T.; Pham, T.T. *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*, 1st ed.; CRC Press: Boca Raton, FL, USA, 2000.
8. Khehra, B.S.; Pharwaha, A.P.S.; Kaushal, M. Fuzzy 2-partition entropy threshold selection based on Big Bang–Big Crunch Optimization algorithm. *Egypt. Inform. J.* **2015**, *16*, 133–150. [[CrossRef](#)]
9. Mahmoodabadi, M.J.; Jahanshahi, H. Multi-objective optimized fuzzy-PID controllers for fourth order nonlinear systems. *Eng. Sci. Technol. Int. J.* **2016**, *19*, 1084–1098. [[CrossRef](#)]
10. Valdez, F.; Melin, P.; Castillo, O. An improved evolutionary method with fuzzy logic for combining Particle Swarm Optimization and Genetic Algorithms. *Appl. Soft Comput.* **2011**, *11*, 2625–2632. [[CrossRef](#)]
11. David, R.C.; Precup, R.E.; Petriu, E.M.; Rădac, M.B.; Preitl, S. Gravitational search algorithm-based design of fuzzy control systems with a reduced parametric sensitivity. *Inf. Sci.* **2013**, *247*, 154–173. [[CrossRef](#)]
12. Vrkalovic, S.; Lunca, E.C.; Borlea, I.D. Model-Free Sliding Mode and Fuzzy Controllers for Reverse Osmosis Desalination Plants. *Int. J. Artif. Intell.* **2018**, *16*, 208–222.
13. Muthiah-Nakarajan, V.; Noel, M.M. Galactic Swarm Optimization: A new global optimization metaheuristic inspired by galactic motion. *Appl. Soft Comput.* **2016**, *38*, 771–787. [[CrossRef](#)]
14. Bernal, E.; Castillo, O.; Soria, J.; Valdez, F. *Galactic Swarm Optimization with Adaptation of Parameters Using Fuzzy Logic for the Optimization of Mathematical Functions*; Springer: Cham, Switzerland, 2018; pp. 131–140.
15. Bernal, E.; Castillo, O.; Soria, J. A fuzzy logic approach for dynamic adaptation of parameters in galactic swarm optimization. In Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 6–9 December 2016; pp. 1–7.
16. Peraza, C.; Valdez, F. Melin, Optimization of Intelligent Controllers Using a Type-1 and Interval Type-2 Fuzzy Harmony Search Algorithm. *Algorithms* **2017**, *10*, 82. [[CrossRef](#)]
17. Bernal, E.; Castillo, O.; Soria, J.; Valdez, F. A variant to the dynamic adaptation of parameters in galactic swarm optimization using a fuzzy logic augmentation. In Proceedings of the 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–7.
18. Olivas, 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.* **2019**, *476*, 159–175. [[CrossRef](#)]
19. Zadeh, L.A. Fuzzy logic = computing with words. *IEEE Trans. Fuzzy Syst.* **1996**, *4*, 103–111. [[CrossRef](#)]
20. Jang, J.S.R.; Sun, C.T.; Mizutani, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*; Prentice Hall: Upper Saddle River, NJ, USA, 1997.
21. Karnik, N.N.; Mendel, J.M.; Liang, Q. Type-2 fuzzy logic systems. *IEEE Trans. Fuzzy Syst.* **1999**, *7*, 643–658. [[CrossRef](#)]

22. Cervantes, L.; Castillo, O.; Soria, J. Hierarchical aggregation of multiple fuzzy controllers for global complex control problems. *Appl. Soft Comput. J.* **2016**, *38*, 851–859. [[CrossRef](#)]
23. Castillo, O.; Aguilar, L.T. *Type-2 Fuzzy Logic in Control of Nonsmooth Systems 373*; Springer International Publishing: Cham, Switzerland, 2019.
24. Liang, Q.; Mendel, J.M. Interval type-2 fuzzy logic systems: Theory and design. *IEEE Trans. Fuzzy Syst.* **2000**, *8*, 535–550. [[CrossRef](#)]
25. Karnik, N.N.; Mendel, J.M. Operations on type-2 fuzzy sets. *Fuzzy Sets Syst.* **2001**, *122*, 327–348. [[CrossRef](#)]
26. Bernal, E.; Castillo, O.; Soria, J.; Valdez, F. Imperialist Competitive Algorithm with Dynamic Parameter Adaptation Using Fuzzy Logic Applied to the Optimization of Mathematical Functions. *Algorithms* **2017**, *10*, 18. [[CrossRef](#)]
27. Melin, P.; Olivas, F.; Castillo, O.; Valdez, F.; Soria, J.; Valdez, M. Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic. *Expert Syst. Appl.* **2013**, *40*, 3196–3206. [[CrossRef](#)]
28. 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**, *2018*, 1–16. [[CrossRef](#)]
29. Mendel, J. Uncertain rule-based fuzzy logic system: Introduction and new directions. *IEEE Comput. Intell. Mag.* **2001**, *2*, 72–73.
30. Castro, J.R.; Castillo, O.; Melin, P. An Interval Type-2 Fuzzy Logic Toolbox for Control Applications. In Proceedings of the 2007 IEEE International Fuzzy Systems Conference, London, UK, 23–26 July 2007; pp. 1–6.
31. Castro, J.R.; Castillo, O.; Melin, P.; Martínez, L.G.; Escobar, S.; Camacho, I. Building Fuzzy Inference Systems with the Interval Type-2 Fuzzy Logic Toolbox. In *Analysis and Design of Intelligent Systems Using Soft Computing Techniques*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 53–62.
32. Amador-Angulo, L.; Mendoza, O.; Castro, J.; Rodríguez-Díaz, A.; Melin, P.; Castillo, O. Fuzzy Sets in Dynamic Adaptation of Parameters of a Bee Colony Optimization for Controlling the Trajectory of an Autonomous Mobile Robot. *Sensors* **2016**, *16*, 1458. [[CrossRef](#)] [[PubMed](#)]
33. 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*; Springer: Cham, Switzerland, 2018; pp. 569–579.
34. 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.* **2015**, *64*, 137–141. [[CrossRef](#)]
35. 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.* **2001**, *9*, 305–318.
36. Lagunes, M.L.; Castillo, O.; Soria, J. *Optimization of Membership Function Parameters for Fuzzy Controllers of an Autonomous Mobile Robot Using the Firefly Algorithm*; Springer: Cham, Switzerland, 2018; pp. 199–206.
37. 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.* **2009**, *179*, 2158–2174. [[CrossRef](#)]
38. Kanayama, Y.; Kimura, Y.; Miyazaki, F.; Noguchi, T. A Stable Tracking Control Method for an Autonomous Mobile Robot. In Proceedings of the IEEE International Conference on Robotics and Automation, Cincinnati, OH, USA, 13–18 May 1990; pp. 384–389.
39. Astudillo, L.; Melin, P.; Castillo, O. *Chemical Optimization Algorithm for Fuzzy Controller Design*; Springer International Publishing: Berlin/Heidelberg, Germany, 2014.
40. Castillo, O.; Amador-Angulo, L.; Castro, J.R.; Garcia-Valdez, M. A comparative study of type-1 fuzzy logic systems, interval type-2 fuzzy logic systems and generalized type-2 fuzzy logic systems in control problems. *Inf. Sci.* **2016**, *354*, 257–274. [[CrossRef](#)]
41. Peraza, C.; Valdez, F.; Garcia, M.; Melin, P.; Castillo, O. A New Fuzzy Harmony Search Algorithm Using Fuzzy Logic for Dynamic Parameter Adaptation. *Algorithms* **2016**, *9*, 69. [[CrossRef](#)]

