

Article

An Enhanced Fuzzy Hybrid of Fireworks and Grey Wolf Metaheuristic Algorithms

Juan Barraza , Luis Rodríguez, Oscar Castillo , Patricia Melin  and Fevrier Valdez * 

Tijuana Institute of Technology, TecNM, Calzada Tecnológico S/N, Tijuana 22414, BC, Mexico; juan.barraza@cenyca.edu.mx (J.B.); luis.rodriguez@tectijuana.edu.mx (L.R.); ocastillo@tectijuana.mx (O.C.); pmelin@tectijuana.mx (P.M.)

* Correspondence: fevrier@tectijuana.mx

Abstract: This research work envisages addressing fuzzy adjustment of parameters into a hybrid optimization algorithm for solving mathematical benchmark function problems. The problem of benchmark mathematical functions consists of finding the minimal values. In this study, we considered function optimization. We are presenting an enhanced Fuzzy Hybrid Algorithm, which is called Enhanced Fuzzy Hybrid Fireworks and Grey Wolf Metaheuristic Algorithm, and denoted as EF-FWA-GWO. The fuzzy adjustment of parameters is achieved using Fuzzy Inference Systems. For this work, we implemented two variants of the Fuzzy Systems. The first variant utilizes Triangular membership functions, and the second variant employs Gaussian membership functions. Both variants are of a Mamdani Fuzzy Inference Type. The proposed method was applied to 22 mathematical benchmark functions, divided into two parts: the first part consists of 13 functions that can be classified as unimodal and multimodal, and the second part consists of the 9 fixed-dimension multimodal benchmark functions. The proposed method presents better performance with 60 and 90 dimensions, averaging 51% and 58% improvement in the benchmark functions, respectively. And then, a statistical comparison between the conventional hybrid algorithm and the Fuzzy Enhanced Hybrid Algorithm is presented to complement the conclusions of this research. Finally, we also applied the Fuzzy Hybrid Algorithm in a control problem to test its performance in designing a Fuzzy controller for a mobile robot.



Citation: Barraza, J.; Rodríguez, L.; Castillo, O.; Melin, P.; Valdez, F. An Enhanced Fuzzy Hybrid of Fireworks and Grey Wolf Metaheuristic Algorithms. *Axioms* **2024**, *13*, 424. <https://doi.org/10.3390/axioms13070424>

Academic Editor: Salvatore Sessa

Received: 26 April 2024

Revised: 17 June 2024

Accepted: 18 June 2024

Published: 24 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: GWO; FWA; hybridization; metaheuristics; packs; wolves; explosion amplitude; fuzzy logic; interval type 2 fuzzy inference system

MSC: 03B52

1. Introduction

Evolutionary Computation (EC), Artificial Neural Networks (ANN), and Fuzzy Logic (FL) are three powerful branches of the computer sciences. Computational intelligence focuses on the study of adaptive mechanisms to allow the performance of complex and dynamic systems, and, as we already know, there are computational systems that imitate nature and human linguistic reasoning.

For this research work, we focus on the Optimization Algorithms [1–3], which are a branch of Evolutionary Computation. Over the years, research scientists have contributed to a wide variety of Optimization Algorithms, especially in the algorithms based on swarm intelligence [4], such as the Grey Wolf Optimizer (GWO) [5] and Fireworks Algorithm (FWA) [6,7]; the previously mentioned algorithms are the basis for the present investigation, in which we added the use of Fuzzy Logic.

Computational Intelligence is a very broad field in the Computational Sciences, but one specific branch is Fuzzy Logic [8], which is a highly researched branch, and this is reflected in different recent applications. One of the applications of Fuzzy Logic is controlling

variables, parameters, or processes. In this case, we are using Fuzzy Logic to dynamically adjust parameters in the Optimization Algorithms.

On the other hand, hybridization has gained strength in the field of computer sciences. We could divide the hybridization into two parts: soft and strong. Soft hybridization is the combination of two or more similar computing systems. On the contrary, the combination of two or more different computing systems can be called strong hybridization [9].

The impact of utilizing Fuzzy Logic in the Optimization Algorithms is shown and detailed in this paper, which is the key contribution of the research. In this work, we are improving the performance of FWA-GWA [10] by introducing a Fuzzy Inference System (FIS) of a Mamdani Type. This is done with the objective of dynamically adjusting the amplitude coefficient, and then, the equilibrium between the exploration and exploitation during the execution of the Hybrid Algorithm is achieved.

Also, it is important to highlight that the present research work is a continuation of previous work, i.e., the hybridization between FWA and GWO algorithms was a novel model that unleashed the development of similar hybridization works. Therefore, the main goal and contribution mentioned in the above paragraph is that the improvement to the hybrid method is by using Fuzzy Logic, in this case, in order to adjust the parameters in the hybrid method (to combine the advantages of both analyzed methods: FWA and GWO). The conventional hybrid has better performance only in the mathematical models of the algorithms and in this case, we are introducing Fuzzy Logic in the mathematical model proposed previously in order to improve even more the performance of the hybrid method.

The content of the paper is organized as follows: a review of FWA-GWO, which is the Conventional Hybrid Algorithm, is shown in Section 2. The proposed EF-FWA-GWO is outlined in Section 3. Section 4 describes the experiments and results. Lastly, Section 5 outlines the conclusions.

2. Conventional FWA-GWO Hybrid Algorithm

It is relevant to mention that in the conventional FWA-GWO (that is, the hybridization of the Fireworks Algorithm and Grey Wolf Optimization), the idea was to mathematically incorporate the best equations into one optimization algorithm, and we briefly review these ideas in the following lines.

The first method is the Fireworks Algorithm (FWA), which was implemented in 2010 by Ying Tan [11,12]. This method is based on the behavior of the fireworks when it explodes generating sparks that are represented in equations. In the hybridization, the authors take advantage of the following features:

- The way the algorithm initialization occurs;
- Explosion amplitude that in the algorithm we can note as the exploration and exploitation modes;
- Finally, the different locations that the sparks can localize in the search range of the optimization problem.

The second algorithm that was selected is the Grey Wolf Optimizer (GWO) that Mirjalili published in 2014 and has thousands of citations around the world; the main inspiration was in the research of Muro [13] about the behavior of *Canis lupus* and the advantages that the hybridization take are the following:

- The feature that the algorithm has to select the best solutions in one hierarchical pyramid;
- The interaction among the population in relationship with the leaders of the algorithm or best solutions;
- The way that the population can move to the next location in the search space based on the hierarchical pyramid.

In order to show the general flow chart and behavior of the conventional FWA-GWO method [14], presented in Figure 1, it is important to mention that the red blocks

represent the features mentioned above about the FWA method, and the blue blocks used the methodology of the GWO algorithm [15].

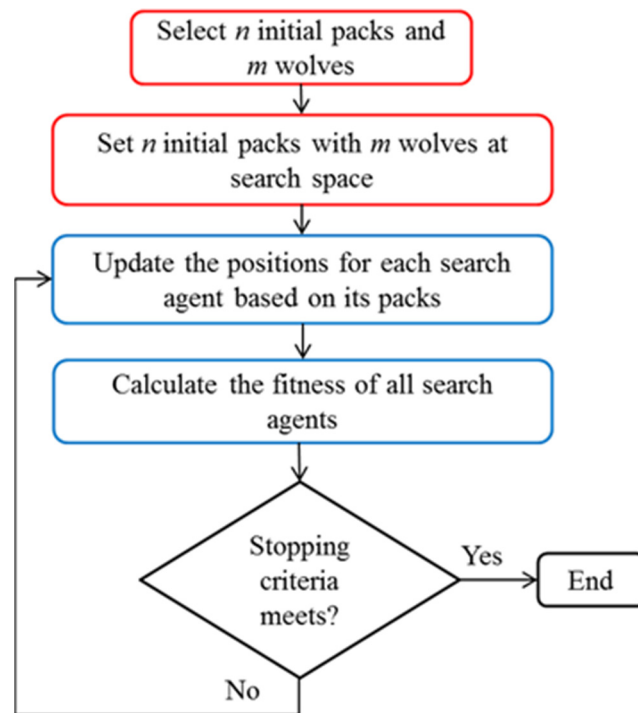


Figure 1. Flow chart of conventional FWA-GWO.

In order to describe mathematically the hybrid method, we are presenting the main equations:

The first step into the algorithm is to determine the number of iterations that the algorithm has based on the function evaluations (O), the number of packs in the algorithm (n), and finally, the number of wolves for each pack at the beginning of the algorithm (m), we can then note these features in Equation (1).

$$T = \frac{O}{(n * m)} \quad (1)$$

It is important to consider the search space that the problem has in order to initialize it in a correct way; in this case, Equations (2)–(4) represent these important features in the hybridization method to determine the locations of each wolf pack (n).

$$TS = ub - lb \quad (2)$$

where ub is the upper boundary of the problem, lb represents the lower boundary, and finally, TS denotes the total search space in the algorithm.

With the value of TS , the algorithm selects the total number of subspaces that it needs according to the number of packs at the beginning (Equation (3)).

$$SS = \frac{TS}{n} \quad (3)$$

With the value of SS , the algorithm can select different subspaces for each wolf pack, and we can note it as a linear function in Equation (4).

$$f(n) = an + b \quad (4)$$

With the value of SS , the algorithm can select different subspaces for each wolf pack, and we can note it as a general linear function in Equation (4). Finally, in Figure 2, we can find an example of this mathematical model.

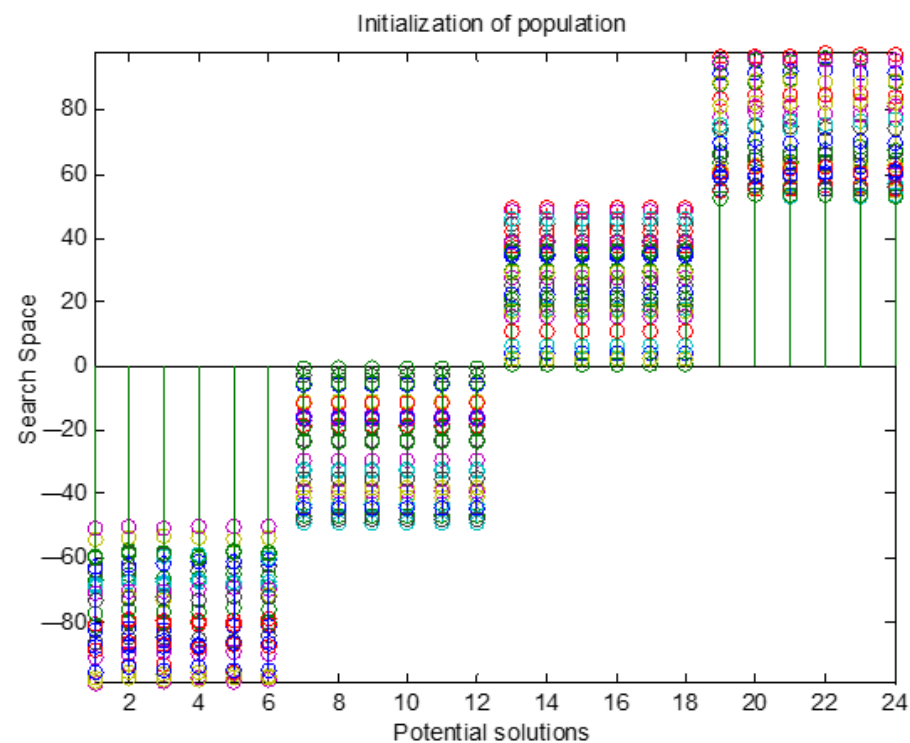


Figure 2. Example of initialization with 4 packs.

Figure 2 shows the example of the initialization of four packs with six wolves for each pack; each pack has an independent search space in order to initialize in different spaces, respectively; in this case, the first pack has initialized space between -100 and -50 , the second pack have initialized space between -50 and 0 , in the third pack wolves have a range space between 0 and 50 , and finally, the last pack (for this example) initialize between 0 and 50 . The figure also represents a problem with 30 dimensions that are the circles for each possible solution; the x -axis represents the number of wolves in the total population, and the y -axis represents the total search space in this example, which is between -100 and 100 .

It is very relevant to highlight that one of the features that performs the hybridization is that the number of packs decreases during iterations of the algorithm; the main goal is to finish with only one bigger pack. Also, the FWA-GWO has the equation to determine the number of packs according to its three phases; we can note the feature in Equation (5) and the graphical way in Table 1.

$$\underline{n} = \frac{n}{2} \quad (5)$$

Table 1. Number of packs during different phases.

Phase		Number of Packs					
1	2	3	4	5	6	7	8
2	1	1	2	2	3	3	4
3	1	1	1	1	1	1	1

These are the features that the FWA-GWO algorithm used in order to analyze the optimization problem. We applied the hierarchical pyramid and its inspiration to take

advantage of both great algorithms. In the next equations, the authors mathematically represent the model and parameters used in the hybridization algorithm.

The first important behavior of the hybridization is the ability to specifically have a search space for each pack of wolves; in this case, it is represented by Equation (6):

$$A_n = \begin{cases} n = 1, & A_1 = 0.5 \\ n = 2, & A_1 = 1 \text{ and } A_2 = 2 \\ n \geq 3, & A_n = \hat{A} \cdot \frac{f(x_n) - y_{min} + \epsilon}{\sum_{i=1}^n (f(x_i) - y_{min}) + \epsilon} \end{cases} \quad (6)$$

This is an important and determinant equation in order to have exploration and exploitation in the algorithm; in order to apply different metaheuristics, the author normalized the mathematical model, as we can note in Equation (7):

$$\hat{A}_n = \frac{2 * A_n}{\text{Max}(A_n)} \quad (7)$$

With these amplitudes, we can calculate the distance between the leader wolves and the rest of the wolves in the same pack; we can note this feature in Equation (8).

$$\vec{D}_n = \left| \hat{A}_n \cdot \vec{X}_{pn}(t) - \vec{X}_n(t) \right| \quad (8)$$

This distance is needed to update the position of each wolf in the next iteration using Equation (9).

$$\vec{X}_n(t+1) = \vec{X}_{pn}(t) - E_n \vec{D}_n \quad (9)$$

where E_n represents the randomness in the algorithm that can be used with any method (Gaussian, levy flights, sine wave, etc.) and is represented in Equation (10).

$$\vec{E}_n = 2a \cdot r_{1n} \quad (10)$$

Finally, it is important to recall that in this method, we used the hierarchy in the best solutions, so these equations applied for the best three solutions in each pack (alpha, beta, and delta), as can be noted in Equations (11), (12), and (13), respectively.

$$\vec{D}_{\alpha n} = \|\vec{C}_{1n} \cdot \vec{X}_{\alpha n} - \vec{X}_n\|, \quad \vec{D}_{\beta} = \|\vec{C}_{2n} \cdot \vec{X}_{\beta n} - \vec{X}_n\|, \quad \vec{D}_{\delta} = \|\vec{C}_{3n} \cdot \vec{X}_{\delta n} - \vec{X}_n\| \quad (11)$$

$$\vec{X}_{1n} = \vec{X}_{\alpha n} - \vec{E}_{1n} \cdot (\vec{D}_{\alpha n}), \quad \vec{X}_{2n} = \vec{X}_{\beta n} - \vec{E}_{2n} \cdot (\vec{D}_{\beta n}), \quad \vec{X}_{3n} = \vec{X}_{\delta n} - \vec{E}_{3n} \cdot (\vec{D}_{\delta n}) \quad (12)$$

$$\vec{X}_n(t+1) = \frac{\vec{X}_{1n} + \vec{X}_{2n} + \vec{X}_{3n}}{3} \quad (13)$$

where Equation (13) is the average of the results among the alpha, beta, and delta solutions, respectively, and C represents the weight that each solution has over iterations that we can denote in Equation (14).

$$\vec{C}_n = \hat{A}_n \cdot r_{2n} \quad (14)$$

In order to represent the mathematical model described above in a graphical way, we can note the pseudocode of the hybridization method in Figure 3.

```

Initialize  $n$  and  $m$ 

Initialize the grey wolf population  $X_{ni}$  ( $i = 1, 2 \dots n$ )

Initialize  $a$ ,  $E_n$  and  $\widehat{A}_n$ 

Calculate the fitness of each search agent

 $X_{n\alpha}$  = the best search agent of  $n$  pack

 $X_{n\beta}$  = the second best agent of  $n$  pack

 $X_{n\delta}$  = the third best search agent of  $n$  pack

while ( $t < \text{Max number of iterations}$ )

    for each search agent

        Update the position of the current
        search agent by Equation (14)

    end for

    Update  $a$ ,  $\widehat{A}_n$  and  $E_n$ 

    Update  $n$  and  $m$ 

    Calculate the fitness of all search agents

    Update  $X_{n\alpha}$ ,  $X_{n\beta}$  and  $X_{n\delta}$ 

     $t = t + 1$ 

```

Figure 3. Pseudocode of conventional FWA-GWO.

3. Proposed Fuzzy Enhanced Hybrid Algorithm

An improvement to the conventional FWA-GWO is presented in this section, which we have named Enhanced Hybrid Fuzzy Fireworks and Grey Wolf Metaheuristic Algorithm (EF-FWA-GWO).

We have implemented a dynamical adjustment of parameters using a Mamdani-type Fuzzy Inference System [16,17] into the Hybrid Algorithm (FWA-GWO). The goal of the FIS is to control the exploration and exploitation during the execution of the algorithm.

The four general steps of the proposed EF-FWA-GWO are enlisted as follows:

1. Select initial locations and wolves;
2. Set initial packs with wolves at search space;
3. Update the positions for each search agent based on its packs;
 - a. Dynamically adjustment parameter with Fuzzy Inference System of a Mamdani type;
4. Calculate the fitness of all search agents.

As we can see, the improvement mentioned in the above paragraph is described in step 3. For a detailed explanation, we have added a flow chart and an algorithm. Figure 4 shows it step by step in a graphic way, and Algorithm 1 shows the logical sequence of the EF-FWA-GWO algorithm.

Algorithm 1: Enhanced Fuzzy Hybrid FWA-GWO

```

Initialize  $n$  and  $m$ 
Initialize the grey wolf population  $X_{ni}$  ( $i = 1, 2 \dots n$ )
Initialize  $a$ ,  $E_n$  and  $H\hat{F}A_n$ 
Calculate the fitness of each search agent
 $X_{n\alpha}$  = the best search agent of  $n$  pack
 $X_{n\beta}$  = the second-best agent of  $n$  pack
 $X_{n\delta}$  = the third best search agent of  $n$  pack
while ( $t < \text{Max number of iterations}$ )
    for each search agent
        Update the position of the current search agent by Equation (14)
    end for
    Update  $H\hat{F}A$  or  $HFA_n$ 
    Update  $n$  and  $m$ 
    Calculate the fitness of all search agents
    Update  $X_{n\alpha}$ ,  $X_{n\beta}$  and  $X_{n\delta}$ 
     $t = t + 1$ 
end while
return best ( $X_{1\alpha}$ ,  $X_{2\alpha} \dots X_{n\alpha}$ )

```

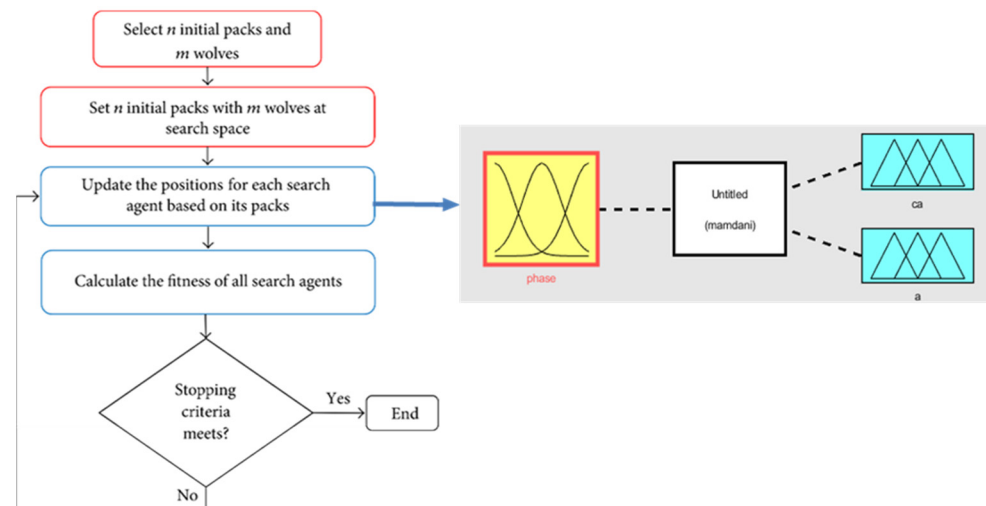


Figure 4. Flow chart of the EF-FWA-GWO.

The EF-FWA-GWO algorithm is a variant of an original work (FWA-GWO), which we have called the Conventional Hybrid Algorithm in this paper. In the original work, we can find a detailed explanation of steps 1, 2, and 4.

Consequently, as we described in step 3, we focus on explaining the dynamical adjustment of parameters with the Fuzzy Inference Systems of a Mamdani Type in this section.

We designed two different Fuzzy Inference Systems (FISs) to implement under a conditional decision. If the conditional was fulfilled, we implemented a Fuzzy Inference System named “FisPacks”; on the contrary, the “FisAmplitude” Fuzzy Inference System [18] was implemented.

The conditional decision to implement the FISs is shown in Figure 5.

We decided to use the conditional described in Figure 5 depending on the phase of the execution algorithm, i.e., in the initial and initial-medium phase, the Fuzzy Inference System [19] called “FisPacks” is executed, which has the objective of adjusting the Coefficient Amplitude with a decreasing value from 45 to 0. The FIS called “FisAmplitude” is executed in the medium-final and the final phase of the Fuzzy Hybrid Algorithm work. The main objective of this Fuzzy Inference System is to adjust the fuzzy exploitation amplitude with a decreasing normalized value from 2 to 0. The above-mentioned FISs allow the adjustment

of parameters into the hybrid FWA-GWO algorithm to achieve the main goal of controlling the exploration and exploitation of the algorithm.

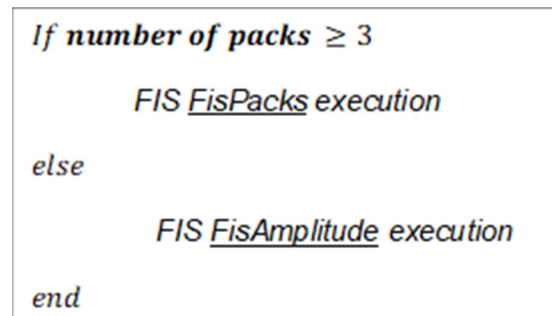


Figure 5. Conditional decision for implementation.

3.1. “FisPacks” Fuzzy Inference Systems

If the “FisPacks” is activated, the principal parameter to control the explosion amplitude of the agents based on its packs is the Amplitude Coefficient; therefore, the parameter that we modify into the Conventional Algorithm is the Amplitude Coefficient. For parameter modification, we are implementing a fuzzy adjustment with the goal of achieving a balance between exploration and exploitation.

The mathematical modification is presented in Equation (15):

$$HFA_n = H\hat{F}A \cdot \frac{f(x_n) - y_{min} + \epsilon}{\sum_{i=1}^n (f(x_n) - y_{min}) + \epsilon} \quad (15)$$

where ($H\hat{F}A$) will be in an increased range between 45 and 0.

As we mentioned above, we have used the FIS of a Mamdani Type [20] to dynamically adjust the parameters. We have decided to implement two variants in the Fuzzy Inference Systems, i.e., we have varied MFs in the input and output variables. The variants are Triangular and Gaussian membership functions.

Figure 6 depicts a Triangular representation of the “FisPacks” Fuzzy Inference System that was used.

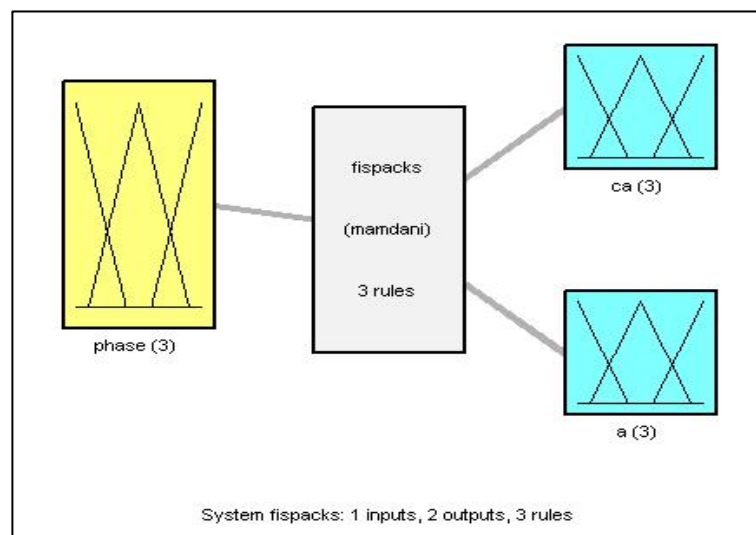


Figure 6. Triangular FIS “FisPacks”.

All Fuzzy Inference Systems [21] variants are conformed by one input and two output variables, as can be observed in Figure 6.

In Figure 7, we are showing the “FisPacks” Gaussian Fuzzy Inference System.

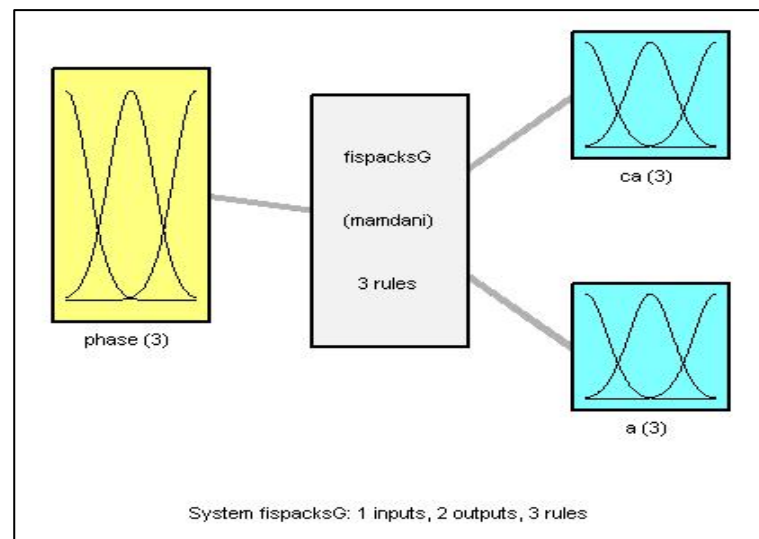


Figure 7. Gaussian FIS “FisPacks”.

The Phase input variable was normalized with values in the range between 0 and 1; the variable has three partitions: low, medium, and high, with the values range depending on the variant of the FIS. This input variable is used for both Fuzzy Inference implementations.

On the contrary, we have implemented two output variables in the Fuzzy Inference System; the first output variable is named *ca* (amplitude coefficient) with a range of values from 45 to 2, and *a* (amplitude) is the second output variable, with a normalized value from 2 to 0. Both output variables have three partitions: low, medium, and high, and, as we mentioned above, the ranges depend on the variant of the Fuzzy Inference System. Below, we show the variants in the membership functions (MFs).

3.1.1. Triangular Membership Functions

The input variable with Triangular MFs is shown in Figure 8.

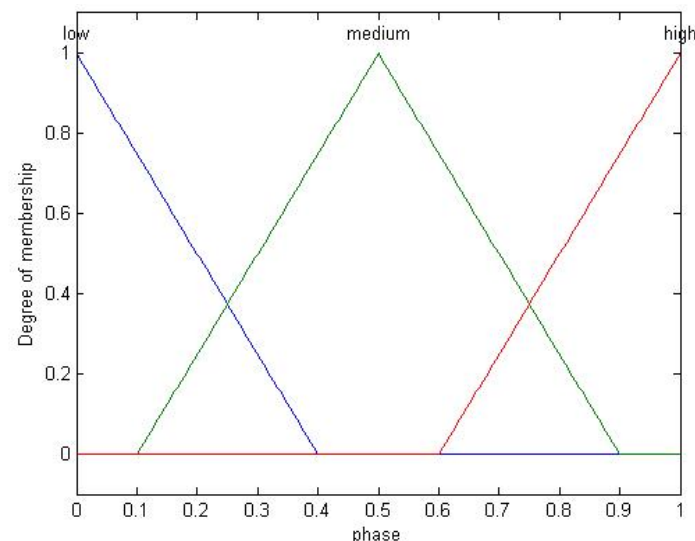


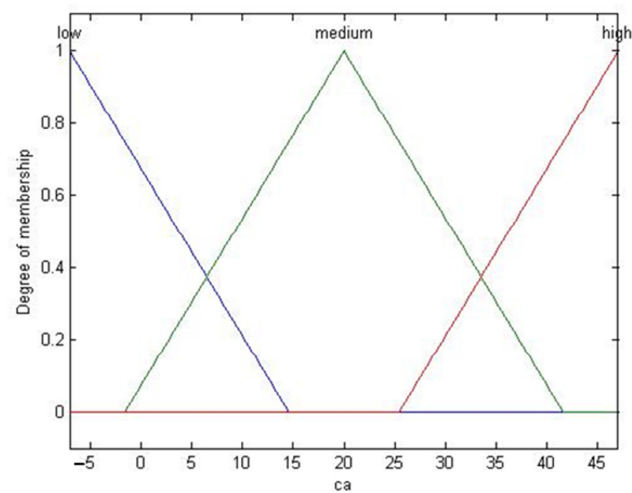
Figure 8. Triangular Input Variable “Phase” in the FIS “FisPacks”.

The range of each membership function in the Phase is listed in Table 2.

Table 2. Triangular Input Variable “Phase” parameters.

Linguistic Values	Parameters
Low	$[-0.4, 0, 0.4]$
Medium	$[0.1, 0.5, 0.9]$
High	$[0.6, 1, 1.4]$

In Figure 9, we show the first output variable (ca) with Triangular membership functions.

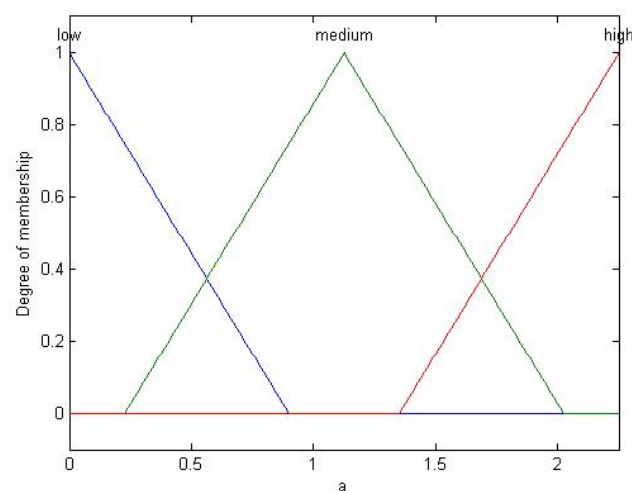
**Figure 9.** Triangular Output Variable “ca” in the FIS “FisPacks”.

The range of each membership function in the ca is listed in Table 3 below.

Table 3. Triangular Output Variable “ca” parameters.

Linguistic Values	Parameters
Low	$[-28.6, -7, 14.6]$
Medium	$[-1.6, 20, 41.6]$
High	$[25.4, 47, 68.6]$

The second output variable (a) with Triangular MFs is shown in Figure 10.

**Figure 10.** Triangular Output Variable “a” in the FIS “FisPacks”.

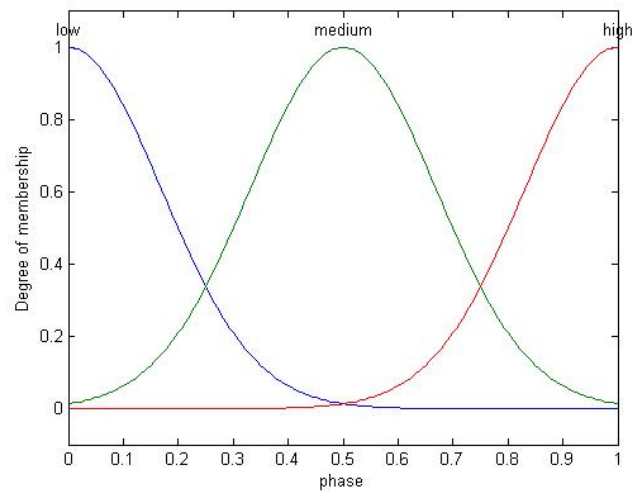
The range of each MF in the “a” variable is listed in Table 4.

Table 4. Triangular Output Variable “a” parameters for first FIS.

Linguistic Values	Parameters
Low	[−0.9, 0, 0.9]
Medium	[0.225, 1.125, 2.025]
High	[1.35, 2.25, 3.15]

3.1.2. Gaussian Membership Functions

The input variable “Phase” with Gaussian membership functions is shown in Figure 11.

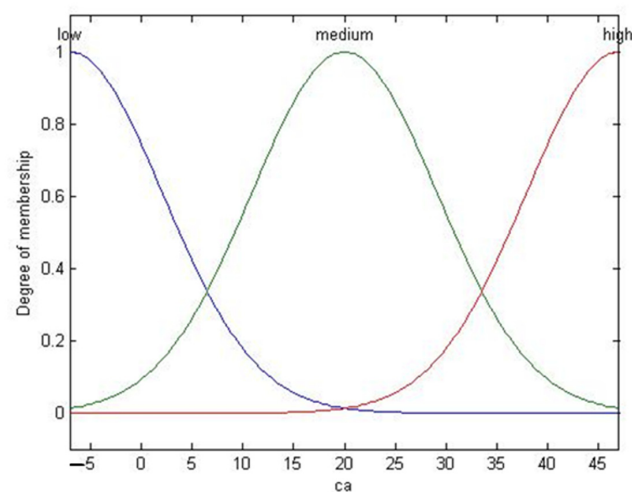
**Figure 11.** Gaussian Input Variable “Phase” in the FIS “FisPacks”.

The range of each membership function in the Phase is listed in Table 5.

Table 5. Gaussian Output Variable “phase” parameters.

Linguistic Variables	Parameters
Low	$[0.1699, 6.93 \times 10^{-18}]$
Medium	[0.1699, 0.5]
High	[0.1699, 1]

The output variables with Gaussian MFs are illustrated in Figures 12 and 13.

**Figure 12.** Gaussian Input Variable “ca” in the FIS “FisPacks”.

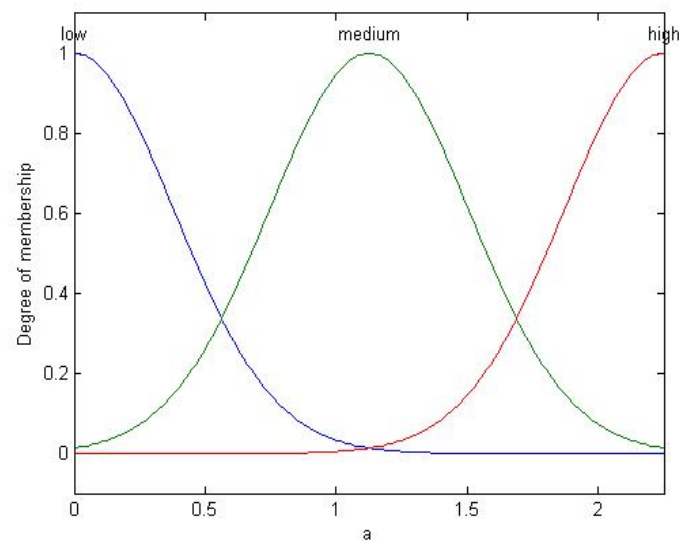


Figure 13. Gaussian Input Variable “a” in the FIS “FisPacks”.

The range of each membership function in the ca is listed in Table 6 below.

Table 6. Gaussian Output Variable “ca” parameters.

Linguistic Values	Parameters
Low	[9.173, −7]
Medium	[9.173, 20]
High	[9.173, 40]

In Table 7, the range of each membership function in the a is listed.

Table 7. Gaussian Output Variable “a” parameters.

Linguistic Variables	Parameters
Low	[0.3822, 6.939×10^{-18}]
Medium	[0.3822, 1.125]
High	[0.3822, 2.25]

3.1.3. Fuzzy Rules

To apply a dynamical adjustment with the Fuzzy Inference System, we are using three Fuzzy rules with the goal of obtaining a decrease in control of the parameters. The Fuzzy rules are shown in Figure 14.

1. If (phase is low) then (ca is high) (a is high)
2. If (phase is medium) then (ca is medium) (a is medium)
3. If (phase is high) then (ca is low) (a is low)

Figure 14. Fuzzy rules in FIS “FisPacks”.

3.1.4. Surfaces

Finally, to verify the optimal performance of the FIS, we added the surfaces of the Fuzzy model showing the behavior of the parameters during algorithm execution. The surfaces are presented in Figures 15 and 16.

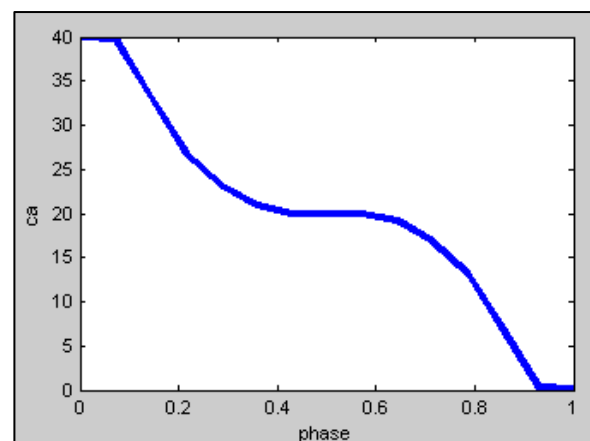


Figure 15. Surface of “ca” parameter of the first FIS.

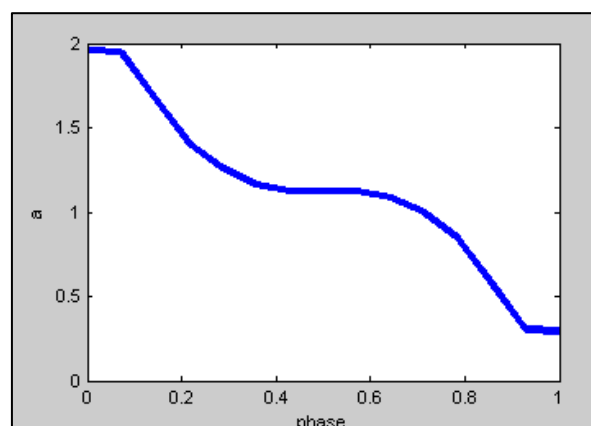


Figure 16. Surface of “a” parameter of the first FIS.

3.2. “FisAmplitude” FIS

For the second implemented FIS (FisAmplitude), the differences in reference to the first FIS (FisPacks) are the output variables. For this case, the output variables are amplitude and a, and both variables have a normalized value from 2 to 0.

In Figures 17 and 18, we show the implementation of the Fuzzy Inference Systems.

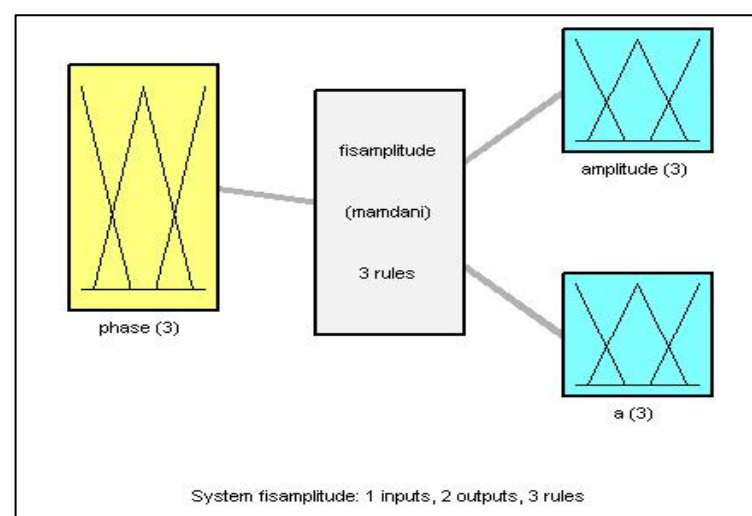


Figure 17. Triangular FIS “FisAmplitude”.

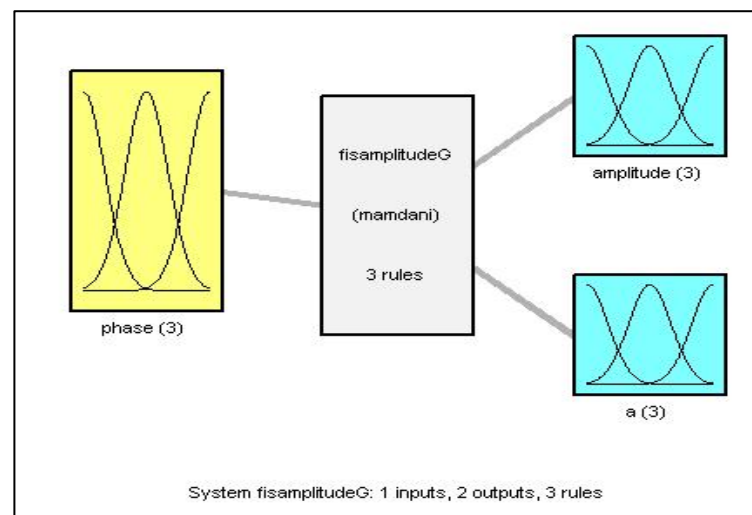


Figure 18. Gaussian FIS "FisAmplitude".

3.2.1. Triangular Membership Functions

In Figures 19 and 20, we show the output variables in the "FispAmplitude" of the implemented FIS.

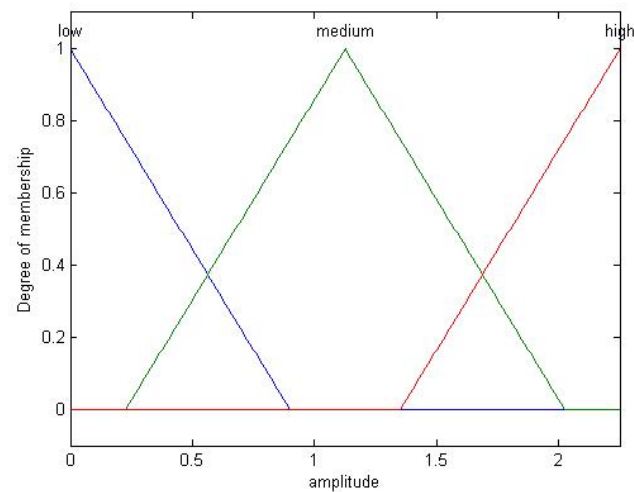


Figure 19. Triangular Output Variable "amplitude" in the FIS "FispAmplitude".

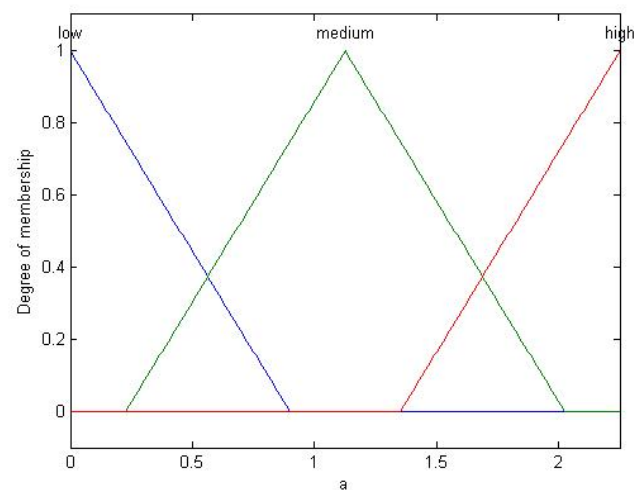


Figure 20. Triangular Output Variable "a" in the FIS "FispAmplitude".

The range of the MFs in the amplitude variable are listed in Table 8.

Table 8. Triangular Output Variable “amplitude” parameters.

Linguistic Values	Parameters
Low	$[-0.9, 0, 0.9]$
Medium	$[0.225, 1.125, 0.025]$
High	$[1.35, 2.25, 3.15]$

The range for the membership functions in the a variable are listed in Table 9.

Table 9. Triangular Output Variable “a” parameter in second FIS.

Linguistic Values	Parameters
Low	$[-0.9, 0, 0.9]$
Medium	$[0.225, 1.125, 0.025]$
High	$[1.35, 2.25, 3.15]$

3.2.2. Gaussian Membership Functions

Figures 21 and 22 show the amplitude and a with their Gaussian membership functions.

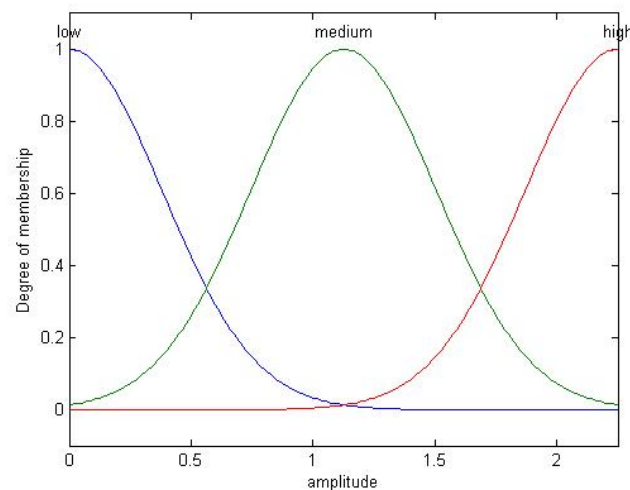


Figure 21. Gaussian Output Variable “amplitude” in the FIS “FispAmplitude”.

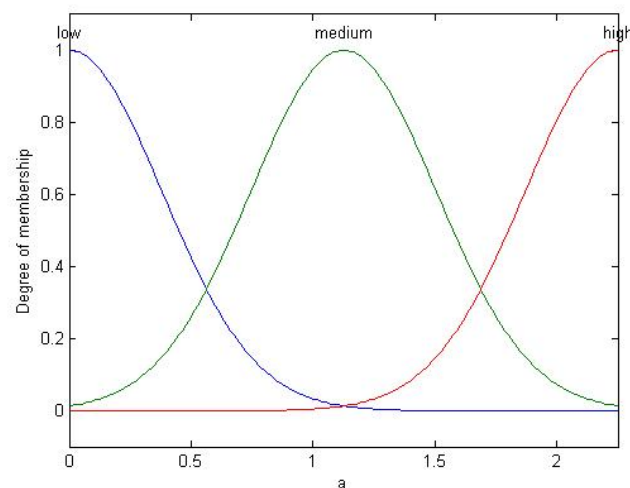


Figure 22. Gaussian Output Variable “a” in the FIS “FispAmplitude”.

In Table 10, the ranges of the Gaussian membership functions in the amplitude are listed.

Table 10. Gaussian Output Variable “amplitude” parameters.

Linguistic Values	Parameters
Low	$[0.3822, 6.939 \times 10^{-18}]$
Medium	$[0.3822, 1.125]$
High	$[0.3822, 2.25]$

The range of membership functions in a are listed in Table 11.

Table 11. Gaussian Output Variable “a” parameters.

Linguistic Variables	Parameters
Low	$[0.3822, 6.939 \times 10^{-18}]$
Medium	$[0.3822, 1.125]$
High	$[0.3822, 2.25]$

3.2.3. Fuzzy Rules

Figure 23 lists the rules used in the “FisAmplitude” FIS.

1. If (*phase* is low) then (*amplitude* is high) (*a* is high)
2. If (*phase* is medium) then (*amplitude* is medium) (*a* is medium)
3. If (*phase* is high) then (*amplitude* is low) (*a* is low)

Figure 23. Fuzzy Rules of “FisAmplitude”.

3.2.4. Surfaces

Decreasing behavior surfaces of the output variables are shown in Figures 24 and 25.

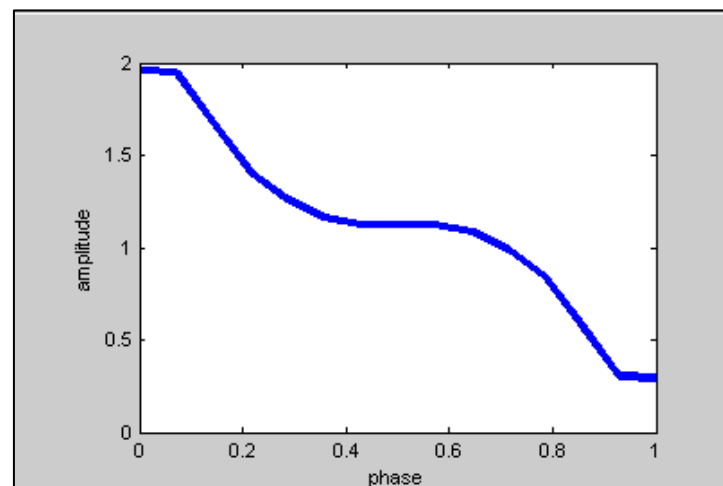


Figure 24. Surface of “amplitude” parameter of the second FIS.

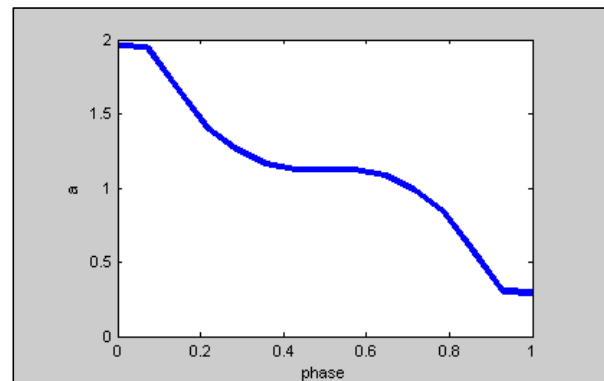


Figure 25. Surface of “a” parameter of the second FIS.

4. Simulation Results and Discussion

In this section, we present the benchmark functions [22–24] that are used in this work in order to verify the improvement of the performance of the hybridization when we used Fuzzy Logic. We also present a statistical summary and a comparison to show the performance of the algorithms and the proposed method.

The first 13 mathematical benchmark functions and their equations are shown in Table 12, which are used for testing the algorithms (Conventional hybridization [25,26] and Fuzzy hybridization [27]). In Figure 26, we are showing the graphical representations in 3D versions of the recently mentioned functions. We can recall that we are not using soft hybridization [28,29], but strong hybridization [30,31].

Table 12. Features of benchmark functions.

Function	Range	$f_{\min}$
$f_1(x) = \sum_{i=1}^n x_i^2$	$[-100, 100]$	0
$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	$[-100, 100]$	0
$f_3(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	$[-100, 100]$	0
$f_4(x) = \max_i \{ x_i , 1 \leq i \leq n \}$	$[-100, 100]$	0
$f_5(x) = \sum_{i=1}^{n-1} \left[100(x_{i+1} - x_i^2)^2 + (x_1 - 1)^2 \right]$	$[-100, 100]$	0
$f_6(x) = \sum_{i=1}^n ([x_i + 0.5])^2$	$[-100, 100]$	0
$f_7(x) = \frac{1}{2} \sum_{i=1}^d (x_i^4 - 16x_i^2 + 5x_i)$	$[-100, 100]$	−39.17
$f_8(x) = \sum_{i=1}^{d/4} \left[(x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4 \right]$	$[-100, 100]$	0
$f_9(x) = \sum_{i=1}^d x_i^2 + \left(\sum_{i=1}^d 0.5ix_i \right)^2 + \left(\sum_{i=1}^d 0.5ix_i \right)^4$	$[-100, 100]$	0
$f_{10}(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-100, 100]$	0
$f_{11}(x) = 20 \exp \left(-0.2 \sqrt{\frac{1}{n}} \sum_{i=1}^n x_i^2 \right) - \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right) + 20 + e$	$[-100, 100]$	0
$f_{12}(x) = \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{y+1})] + (y_n - 1)^2 \right\} + \sum_{i=1}^n u(x_i, 10, 100, 4)$ $y_1 = 1 + \frac{x_1+1}{4}$	$[-100, 100]$	0
$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$		
$f_{13}(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	$[-100, 100]$	0

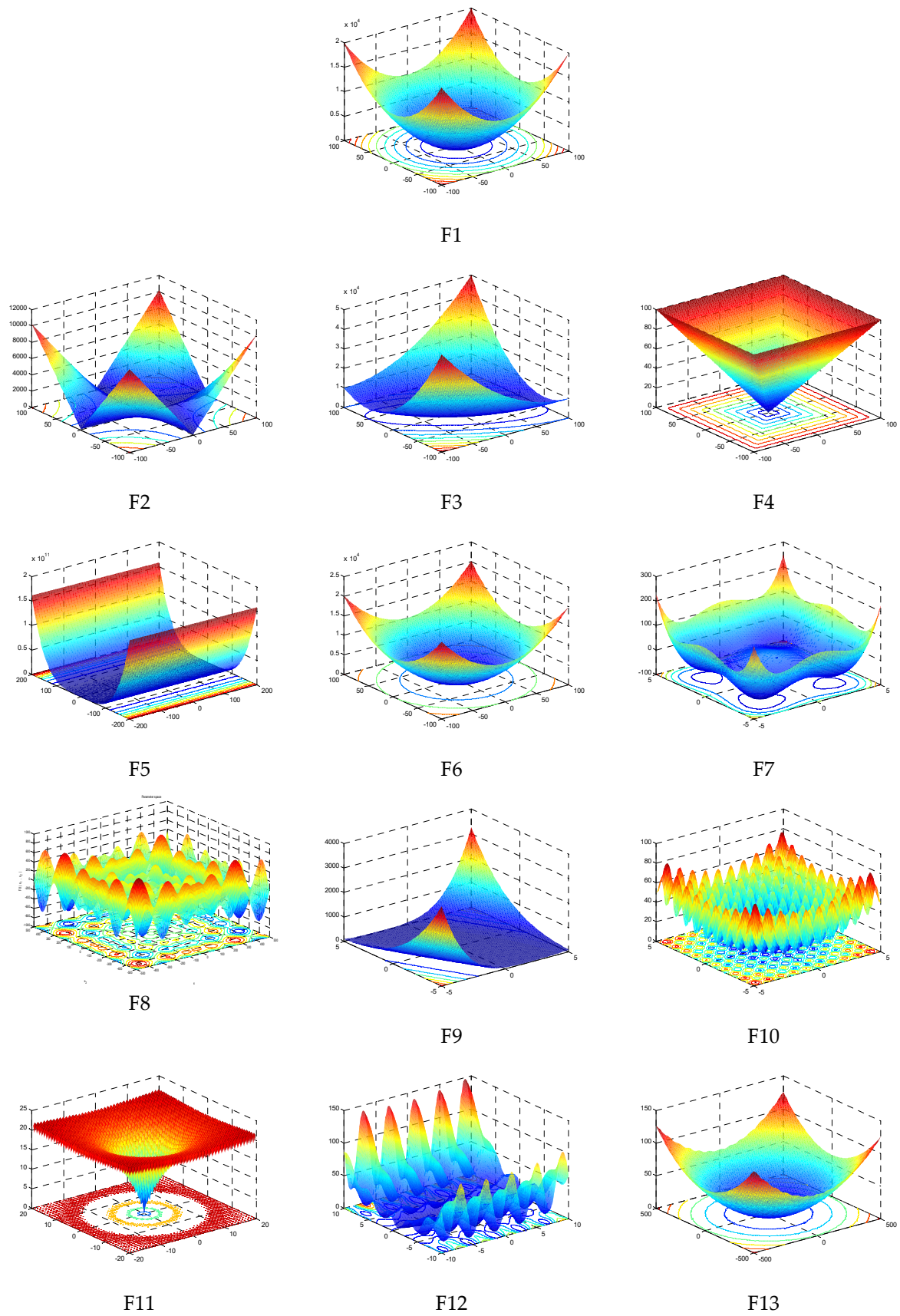
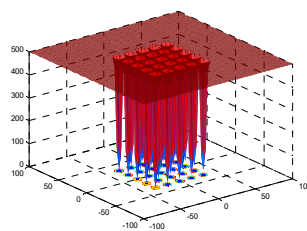


Figure 26. Graphical representations of benchmark functions.

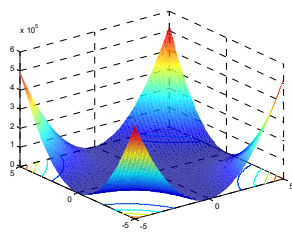
Another set of mathematical benchmark functions is used, which is called fixed-dimension multimodal and contains nine different functions. The equations of the fixed-dimension multimodal are shown in Table 13, and we also show 3D versions of the graphical representations of the mentioned functions in Figure 27.

Table 13. Fixed-dimension multimodal functions.

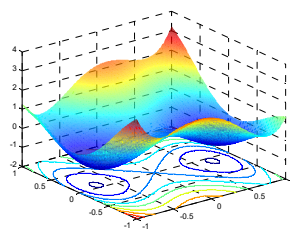
Function	Dim	Range	$f_{\min}$
$f_{14}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=2}^2 (x_i - a_{ij})^6} \right)^{-1}$	2	$[-65, 65]$	1
$f_{15}(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_1(b_1^2 + b_i x_2)}{b_1^2 + b_i x_3 + x_4} \right]^2$	4	$[-5, 5]$	0.00030
$f_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	$[-5, 5]$	−1.0316
$f_{17}(x) = \left[1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2) \right] \times \left[30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2) \right]$	2	$[-2, 2]$	3
$f_{18}(x) = -\sum_{i=1}^4 c_i \exp \left(-\sum_{j=1}^3 a_{ij} (x_j - p_{ij})^2 \right)$	3	$[1, 3]$	−3.86
$f_{19}(x) = -\sum_{i=1}^4 c_i \exp \left(-\sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2 \right)$	6	$[0, 1]$	−3.32
$f_{20}(x) = -\sum_{i=1}^5 \left[(X - a_i)(X - a_i)^T + c_i \right]^{-1}$	4	$[0, 10]$	−10.1532
$f_{21}(x) = -\sum_{i=1}^7 \left[(X - a_i)(X - a_i)^T + c_i \right]^{-1}$	4	$[0, 10]$	−10.4028
$f_{22}(x) = -\sum_{i=1}^{10} \left[(X - a_i)(X - a_i)^T + c_i \right]^{-1}$	4	$[0, 10]$	−10.5363



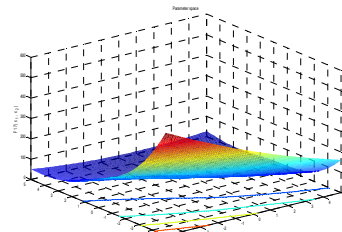
F14



F15



F16



F17

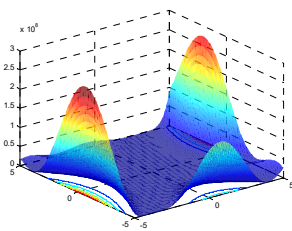
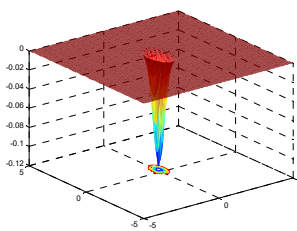
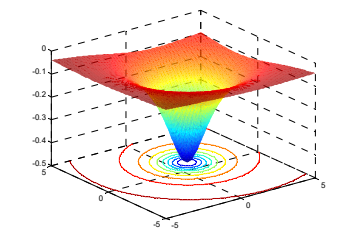


Figure 27. Fixed-dimension multimodal mathematical functions in 3D versions.

The experimentation was realized with the three different configurations that the authors proposed in the conventional hybridization and are the following:

- Version 1: four packs, six wolves for each pack;
- Version 2: five packs, six wolves for each pack;
- Version 3: eight packs, five wolves for each pack.

It is important to mention that in each version, the function evaluations are the same (15,000) in order for the conditions of the methods to be equal. Each algorithm is executed in 32 independent runs to obtain the media and standard deviation. Finally, for comparing the performance of the algorithms, we realized hypothesis tests (Z-test) that we can find in the next tables.

4.1. Comparison between Conventional FWAGWO and F-FWAGWO

In this section, we are summarizing the results that were obtained according to the specifications that we mentioned above. In Table 14, we show the average standard deviation and, finally, the z-value of the hypothesis tests for each function that we analyzed; in this case, for 30 dimensions and version 1, we note that the proposed method is better than the conventional in 3 out of 12 functions, and for the last function, both have the same performance.

Table 14. Comparison of version 1 between conventional FWA GWO and the Fuzzy FWA-GWO in 30 dimensions.

Function	Hybrid V1	STD	Hybrid FV1	STD	Z-Value
F1	6.43×10^{-28}	1.17×10^{-27}	1.1948×10^{-31}	2.41×10^{-31}	−3.9240
F2	1.70×10^{-16}	1.63×10^{-16}	2.13×10^{-18}	2.98×10^{-18}	−7.3536
F3	8.47×10^{-4}	0.0025	3.07×10^{-4}	0.0007	−1.4900
F4	0.0383	0.0584	0.0080	0.0146	−3.5971
F5	27.6283	0.5878	27.5833	0.5480	−0.3996
F6	0.8856	0.4284	0.9831	0.5835	0.9619
F7	−775.8908	98.3576	−751.4547	99.0596	−1.2501
F8	1.7697	6.7631	20.6972	74.5578	1.8055
F9	9641.80	11,406.71	18,071.97	29,673.06	1.8938
F10	1.731	2.2942	2.7775	6.4951	1.0849
F11	5.8606	9.4934	8.7885	10.5125	1.4761
F12	0.0693	0.0297	0.1155	0.0713	4.2727
F13	0	0	0.00	0.00	

In Table 15, we note that in version 2, the Fuzzy method has a better performance in 7 of the 12 benchmarks analyzed.

Table 15. Comparison of version 2 between conventional FWA GWO and the Fuzzy FWA-GWO in 30 dimensions.

Function	Hybrid V2	STD	Hybrid FV2	STD	Z-Value
F1	1.84×10^{-22}	7.49×10^{-22}	8.4637×10^{-26}	2.8773×10^{-25}	−1.7536
F2	1.17×10^{-13}	1.20×10^{-13}	1.84×10^{-15}	1.65×10^{-15}	−6.8525
F3	0.0044	0.0166	4.19×10^{-4}	0.0007	−1.7112
F4	0.0409	0.044	0.0062	0.0083	−5.5262

Table 15. *Cont.*

Function	Hybrid V2	STD	Hybrid FV2	STD	Z-Value
F5	27.5222	0.5276	27.7630	0.7437	1.8863
F6	0.5746	0.3179	0.6856	0.4105	1.5274
F7	−811.8833	98.7647	−846.9459	114.8316	−1.6532
F8	7.55×10^{-5}	1.91×10^{-4}	3.76×10^{-5}	4.51×10^{-5}	−1.3791
F9	15,241.4387	16,021.8731	8564.5959	10,302.3794	−2.5032
F10	1.3627	2.0485	10.3246	19.3669	3.2863
F11	1.71×10^{-11}	4.23×10^{-11}	2.62×10^{-13}	3.28×10^{-13}	−2.8426
F12	0.0705	0.0316	0.1213	0.1184	2.9641
F13	0	0	0.00	0.00	

Finally, for version 3 (Table 16), which has 30 dimensions, the proposed method is better than the conventional method in 8 of the 13 analyzed functions.

Table 16. Comparison of version 3 between conventional FWA GWO and the Fuzzy FWA-GWO in 30 dimensions.

Function	Hybrid V3	STD	Hybrid FV3	STD	Z-Value
F1	1.8×10^{-18}	2.7×10^{-18}	5.7×10^{-21}	3.0×10^{-20}	−4.6926
F2	4.19×10^{-11}	3.52×10^{-11}	4.58×10^{-13}	5.19×10^{-13}	−8.4070
F3	0.0095	0.0241	6.50×10^{-4}	0.0017	−2.6161
F4	0.0248	0.0207	0.0046	0.0038	−6.8395
F5	27.7054	0.5847	27.6202	0.5759	−0.7418
F6	0.59297	0.35281	0.5904	0.3383	−0.0376
F7	−753.7345	291.2693	−981.3364	116.3217	−5.1824
F8	0.1544	0.6904	0.0013	0.0024	−1.5838
F9	6439.18	23,019.40	5305.38	10,361.15	−0.3208
F10	3.1066	3.9244	10.0339	18.1246	2.6677
F11	1.33×10^{-9}	1.34×10^{-9}	2.67×10^{-11}	3.21×10^{-11}	−6.9436
F12	0.0416	0.0249	0.0295	0.0284	−2.2874
F13	9.33×10^{-17}	2.21×10^{-16}	0.00	0.00	−3.0149

In the tables below, we show the three versions presented above, but in this case, the functions have 60 dimensions (the problem is more complex). We note that for 90 dimensions, the proposed method in version 1 has better performance than the conventional method in 6 of the 13 analyzed problems, and the results are presented in Table 17.

Table 17. Comparison of version 1 between conventional FWA GWO and the Fuzzy FWA-GWO in 60 dimensions.

Function	Hybrid V1	STD	Hybrid FV1	STD	Z-Value
F1	6.78×10^{-18}	1.14×10^{-17}	7.92×10^{-20}	3.29×10^{-19}	−4.1959
F2	1.55×10^{-11}	1.28×10^{-11}	3.68×10^{-13}	2.20×10^{-13}	−8.4412
F3	2.3311	2.0824	4.0899	7.5120	1.6113
F4	3.2579	1.1983	2.2917	1.4838	−3.6177
F5	57.9631	0.7311	58.0651	0.6640	0.7376
F6	4.3132	0.7794	5.0155	0.9131	4.1778

Table 17. *Cont.*

Function	Hybrid V1	STD	Hybrid FV1	STD	Z-Value
F7	−1273.7126	157.2159	−1382.53	293.7782	−2.3323
F8	975.1898	2020.46	5260.53	14,248.47	2.1266
F9	123,474.29	105,804.6	137,306.33	147,064.20	0.5452
F10	5.3125	6.2564	9.1421	8.5493	2.5815
F11	13.3142	10.0509	9.5331	10.6785	−1.8413
F12	0.1571	0.1423	0.1323	0.0836	−1.0752
F13	1.04×10^{-16}	3.11×10^{-16}	0.00	0.00	−2.3881

In Table 18, we note that for version 2, the FFWAGWO is better than the FWAGWO in 6 of the 13 functions that we analyzed in this work.

Table 18. Comparison of version 2 between conventional FWA GWO and the Fuzzy FWA-GWO in 60 dimensions.

Function	Hybrid V2	STD	Hybrid FV2	STD	Z-Value
F1	4.61×10^{-14}	1.31×10^{-13}	1.66×10^{-16}	3.18×10^{-16}	−2.5041
F2	2.37×10^{-9}	1.37×10^{-9}	1.36×10^{-10}	8.73×10^{-11}	−11.6228
F3	13.7898	15.3383	10.5559	12.5115	−1.1668
F4	1.1251	0.3469	0.6500	0.3190	−7.1991
F5	57.9556	0.6457	58.0125	0.6017	0.4605
F6	3.8645	0.7918	4.1065	0.8431	1.4942
F7	−1572.6017	164.5944	−1446.4868	181.8827	3.6716
F8	1.9391	13.6862	0.0372	0.2033	−0.9923
F9	97,623.32	73,980.79	161,436.30	194,085.14	2.1940
F10	6.8427	5.1627	16.0706	11.3228	5.2957
F11	3.68×10^{-7}	1.23×10^{-6}	2.377×10^{-8}	8.097×10^{-8}	−1.9943
F12	0.5661	0.4146	0.3825	0.3618	−2.3831
F13	1.07×10^{-14}	2.42×10^{-14}	3.58×10^{-17}	8.30×10^{-17}	−3.1470

Finally, Table 19 shows the z-value of the benchmark functions in 60 dimensions with version 3 of the configuration algorithms, and we note that the proposed method has better performance in 7 of the 13 functions analyzed.

Table 19. Comparison of version 3 between conventional FWA GWO and the Fuzzy FWA-GWO in 60 dimensions.

Function	Hybrid V3	STD	Hybrid FV3	STD	Z-Value
F1	8.17×10^{-11}	1.91×10^{-10}	2.44×10^{-13}	4.59×10^{-13}	−3.0456
F2	1.24×10^{-7}	1.01×10^{-7}	4.79×10^{-9}	2.72×10^{-9}	−8.4257
F3	3.9614	5.0226	2.9745	5.3663	−0.9589
F4	1.2317	0.6411	0.6592	0.4376	−5.2667
F5	58.5643	0.5194	57.9718	0.7475	−4.6488
F6	4.6116	1.0565	3.6102	1.0798	−4.7338
F7	−2020.5185	251.665	−2036.4981	121.8630	−0.4081

Table 19. *Cont.*

Function	Hybrid V3	STD	Hybrid FV3	STD	Z-Value
F8	81.8637	197.114	7.2381	12.4364	−2.6983
F9	231,548.53	432,289.99	197,193.30	269,419.40	−0.4817
F10	21.6702	30.8963	30.6226	38.9509	1.2859
F11	5.10×10^{-6}	9.24×10^{-6}	2.26×10^{-7}	1.87×10^{-7}	−3.7664
F12	0.0723	0.0566	0.0839	0.0610	0.9979
F13	3.14×10^{-11}	1.31×10^{-10}	5.81×10^{-7}	3.23×10^{-6}	1.2826

The benchmark functions, according to the mathematical model, have the option of the problem having “n” dimensions or variables among functions 1 and 13 of the benchmark functions. In order to test the proposal, we increased the number of variables in the problem, and in the following tables, we present the functions with 90 dimensions in order to compare the algorithms when the problems are complex.

In Table 20, we note that in version 1, the Fuzzy method has better performance than the conventional method in 4 out of the 13 benchmarks analyzed.

Table 20. Comparison of version 1 between conventional FWA GWO and the Fuzzy FWA-GWO in 90 dimensions.

Function	Hybrid V1	STD	Hybrid FV1	STD	Z-Value
F1	3.59×10^{-13}	1.05×10^{-12}	1.31×10^{-15}	2.63×10^{-15}	−2.4328
F2	1.71×10^{-9}	9.38×10^{-10}	1.19×10^{-10}	9.03×10^{-11}	−12.0566
F3	170.1456	208.229	190.2772	522.9864	0.2554
F4	9.7444	3.1814	9.6617	3.8646	−0.1179
F5	88.8722	0.4571	88.6116	0.3810	−3.1269
F6	8.869	1.4223	9.0872	1.9827	0.6386
F7	−2958.527	123.562	−2594.0221	281.0649	8.4784
F8	32,391.5762	49,298.17	82,287.85	162,803.28	2.0948
F9	427,794.09	576,474.77	340,907.53	484,611.35	−0.8239
F10	19.1615	20.626	23.8487	21.2635	1.1300
F11	14.8326	9.8082	17.2033	8.2971	1.3178
F12	0.4483	0.5499	0.0737	0.0549	−4.8404
F13	6.11×10^{-12}	3.76×10^{-11}	1.34×10^{-14}	5.34×10^{-14}	−1.1579

The proposed method in version 2 has a better performance than the conventional method in 8 out of the 13 functions analyzed in this work, as shown in Table 21. Finally, for version 3, the FFWAGWO is better than FWAGWO in 6 out of the 13 functions as we can note in Table 22.

Table 21. Comparison of version 2 between conventional FWA GWO and the Fuzzy FWA-GWO in 90 dimensions.

Function	Hybrid V2	STD	Hybrid FV2	STD	Z-Value
F1	3.18×10^{-10}	7.68×10^{-10}	3.63×10^{-12}	4.35×10^{-12}	−2.9194
F2	1.45×10^{-7}	1.02×10^{-7}	1.35×10^{-8}	6.96×10^{-9}	−9.1854
F3	218.3702	141.9908	168.5144	152.0538	−1.7114
F4	3.2032	0.9551	2.5327	0.7804	−3.8825

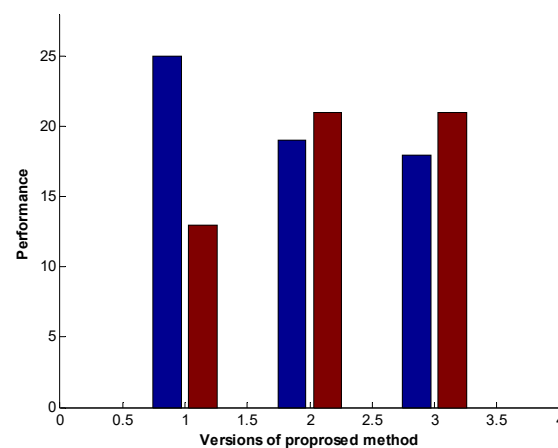
Table 21. *Cont.*

Function	Hybrid V2	STD	Hybrid FV2	STD	Z-Value
F5	89.5622	5.1037	88.4004	0.3822	−1.6212
F6	9.038	1.2308	9.5703	1.2616	2.1568
F7	−2382.765	133.511	−2293.44	162.75	3.0304
F8	1.334	5.1118	0.0456	0.1678	−1.7990
F9	353,505.71	442,838.28	337,757.79	295,953.954	−0.2111
F10	28.204	20.3175	51.6813	41.4278	3.6336
F11	2.01×10^{-4}	4.50×10^{-4}	6.44×10^{-6}	1.11×10^{-5}	−3.0867
F12	1.3641	0.5779	0.7542	0.3633	−6.3806
F13	1.21×10^{-9}	4.38×10^{-9}	2.73×10^{-12}	1.01×10^{-11}	−1.9684

Table 22. Comparison of version 3 between conventional FWA GWO and the Fuzzy FWA-GWO in 90 dimensions.

Function	Hybrid V3	STD	Hybrid FV3	STD	Z-Value
F1	6.02×10^{-7}	9.58×10^{-7}	3.22×10^{-9}	5.05×10^{-9}	−4.4636
F2	4.35×10^{-6}	2.27×10^{-6}	2.86×10^{-7}	1.92×10^{-7}	−12.7412
F3	68.9997	71.6442	64.5839	82.3960	−0.2888
F4	3.6878	1.6607	3.1730	1.5990	−1.5949
F5	90.1465	1.4684	88.5058	0.5688	−7.4406
F6	7.5438	2.503	6.5657	2.0512	−2.1584
F7	−2485.267	427.572	−2287.9810	169.6956	3.0627
F8	397.2208	835.3509	115.9959	179.6233	−2.3505
F9	541,955.02	846,552.82	827,106.07	823,110.85	1.7247
F10	64.8371	63.4859	68.3618	62.0739	0.2835
F11	3.00×10^{-4}	4.00×10^{-4}	4.28×10^{-5}	1.06×10^{-4}	−4.4369
F12	0.2455	0.7294	0.4034	0.5814	1.2092
F13	8.60×10^{-7}	4.95×10^{-6}	3.38×10^{-9}	1.34×10^{-8}	−1.2358

The tests realized when the problem had the flexibility to increase and decrease the number of variables; hence, we analyzed these results and represented them in a graphical way in Figure 28.

**Figure 28.** Results analysis.

The blue bars represent the number of benchmark functions that have better performance than Fuzzy hybridization, and the red bars represent the best performance of the proposed method according to the version or configuration of packs, wolves per pack, and evaluation functions. It is very important to remember that these configurations are the same as proposed in the original work and that we can improve these results with change parameters such as packs and wolves that we can select at the beginning of the algorithm. For this work and according to the Fuzzy proposed method, in general tests, the best versions that improve the results are versions 2 and 3.

Finally, in Tables 23–25, we note the results of the proposed Fuzzy method in the functions that we presented in Table 3 to test and prove that the Fuzzy proposed method works with low dimensions.

Table 23. Comparison of version 1 between conventional FWA-GWO and Fuzzy FWAGWO with fixed dimensions multimodal functions.

Function	Hybrid V1	STD	Hybrid FV1	STD
F14	2.7499	2.8626	3.5215	3.3115
F15	7.42×10^{-4}	2.70×10^{-4}	8.06×10^{-4}	4.55×10^{-4}
F16	−1.0315	5.60×10^{-4}	−1.0316	1.17×10^{-6}
F17	3.0007	8.20×10^{-4}	3.0007	7.60×10^{-4}
F18	−3.8612	0.0024	−3.8611	0.0024
F19	−3.3124	0.0449	−3.3151	0.0213
F20	−5.3093	0.6601	−5.0480	0.0041
F21	−5.2378	0.5854	−5.2016	0.6736
F22	−5.3646	0.4713	−5.2136	0.4032

Table 24. Comparison of version 2 between conventional FWA-GWO and Fuzzy FWAGWO with fixed dimensions multimodal functions.

Function	Hybrid V2	STD	Hybrid FV2	STD
F14	4.1664	3.7214	4.3546	4.1228
F15	7.00×10^{-4}	3.18×10^{-4}	8.43×10^{-4}	4.08×10^{-4}
F16	−1.0316	2.61×10^{-5}	−1.0316	1.98×10^{-4}
F17	3.0009	9.83×10^{-4}	3.0006	6.26×10^{-4}
F18	−3.8612	0.0022	−3.8612	0.0024
F19	−3.3093	0.0484	−3.3045	0.0566
F20	−5.5335	0.9876	−5.0547	0.0414
F21	−5.3205	0.6697	−5.1542	0.2801
F22	−5.5058	0.7676	−5.1389	0.1022

Table 25. Comparison of version 3 between conventional FWA-GWO and Fuzzy FWAGWO with fixed dimensions multimodal functions.

Function	Hybrid V3	STD	Hybrid FV3	STD
F14	4.1664	3.7214	4.3546	4.1228
F15	7.00×10^{-4}	3.18×10^{-4}	8.43×10^{-4}	4.08×10^{-4}
F16	−1.0316	2.61×10^{-5}	−1.0316	1.98×10^{-4}
F17	3.0009	9.83×10^{-4}	3.0006	6.26×10^{-4}

Table 25. *Cont.*

Function	Hybrid V3	STD	Hybrid FV3	STD
F18	−3.8612	0.0022	−3.8612	0.0024
F19	−3.3093	0.0484	−3.3045	0.0566
F20	−5.5335	0.9876	−5.0547	0.0414
F21	−5.3205	0.6697	−5.1542	0.2801
F22	−5.5058	0.7676	−5.1389	0.1022

In order to compare with other recent metaheuristics and hybrid methods, we analyzed the hybrid method that used the same algorithms [32] (Fireworks Algorithm and Grey Wolf Optimizer), and the results are shown in Tables 26–28, respectively. We present the comparative results with the mean and standard deviation of the benchmark functions of both proposed methods.

Table 26. Comparison of version 1 between conventional FWGWO and Fuzzy FWAGWO Version 1 with 60 dimensions in benchmark functions.

Function	FWGWO	STD	Hybrid FV1	STD
F1	5.16×10^{-18}	1.93×10^{-16}	7.92×10^{-20}	3.29×10^{-19}
F2	1.7400	60.1000	4.0899	7.5120
F3	1.02×10^{-3}	3.06×10^{-3}	2.2917	1.4838
F4	90.00	17.40	58.0651	0.6640
F5	3.02×10^{-2}	2.42×10^{-2}	5.0155	0.9131
F9	4.43×10^{-2}	8.31×10^{-2}	9.1421	8.5493
F10	4.21×10^{-10}	5.61×10^{-10}	9.5331	10.6785
F11	1.11×10^{-17}	1.20×10^{-16}	0.00	0.00
F12	2.13×10^{-3}	6.89×10^{-3}	0.1323	0.0836

Table 27. Comparison of version 2 between conventional FWGWO and Fuzzy FWAGWO Version 2 with 60 dimensions in benchmark functions.

Function	FWGWO	STD	Hybrid FV2	STD
F1	5.16×10^{-18}	1.93×10^{-16}	1.66×10^{-16}	3.18×10^{-16}
F2	1.7400	60.1000	10.5559	12.5115
F3	1.02×10^{-3}	3.06×10^{-3}	0.6500	0.3190
F4	90.000	17.400	58.0125	0.6017
F5	0.0302	0.0242	4.1065	0.8431
F9	0.0443	0.0831	16.0706	11.3228
F10	4.21×10^{-10}	5.61×10^{-10}	2.38×10^{-8}	8.10×10^{-8}
F11	1.11×10^{-17}	1.20×10^{-16}	3.58×10^{-17}	8.30×10^{-17}
F12	2.13×10^{-3}	6.89×10^{-3}	0.3825	0.3618

Table 28. Comparison of version 3 between conventional FWGWO and Fuzzy FWAGWO Version 2 with 60 dimensions in benchmark functions.

Function	FWGWO	STD	Hybrid FV3	STD
F1	5.16×10^{-18}	1.93×10^{-16}	2.44×10^{-13}	4.59×10^{-13}
F2	1.7400	60.1000	2.9745	5.3663
F3	1.02×10^{-3}	3.06×10^{-3}	0.6592	0.4376
F4	9.00×10^1	1.74×10^1	57.9718	0.7475
F5	3.02×10^{-2}	2.42×10^{-2}	3.6102	1.0798
F9	4.43×10^{-2}	8.31×10^{-2}	30.6226	38.9509
F10	4.21×10^{-10}	5.61×10^{-10}	2.26×10^{-7}	1.87×10^{-7}
F11	1.11×10^{-17}	1.20×10^{-16}	5.81×10^{-7}	3.23×10^{-6}
F12	2.13×10^{-3}	6.89×10^{-3}	0.0839	0.0610

4.2. Control Problem with EF-FWAGWO

In this subsection, we present and apply the Enhanced Hybrid Fuzzy Fireworks and Grey Wolf Metaheuristic Algorithm to a control problem. The results of this case study were obtained using version 3 of the EF-FWAGWO.

The control problem consists of designing a Fuzzy controller for an autonomous mobile robot system using the proposed method (EF-FWAGWO).

We expect to obtain a good performance of the proposed method in achieving the minimization of the error with respect to a pre-established trajectory in the robot. The Fuzzy Hybrid Algorithm is used to obtain the optimal Fuzzy controller that will be able to provide the control actions to the motors.

In Figure 29, we can find the configuration of the robot.

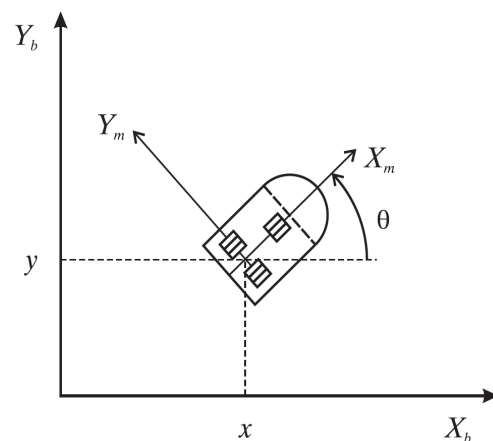


Figure 29. Autonomous mobile robot model.

The mobile robot has two wheels with motors and a passive wheel for the stabilization of the system, and their operation is represented by the two following equations:

$$(q) \dot{v} + C(q, \dot{q})v + Dv = \tau + P(\tau) \quad (16)$$

$$\dot{q} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (17)$$

where $q = (x, y, \theta)^T$ is the coordinate vector, which describes the position of the robot, $v = (v, w)^T$ is the linear and angular velocity vector, $\tau = (\tau_1, \tau_2)$ is the torque vector

applied to the wheels of the robot, where 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 matrix, $C(q, \dot{q})v$ is the vector of centripetal and Coriolis forces, and $D \in R^{2 \times 2}$ is a diagonal positive-defined damping matrix.

In addition, the FIS of the mobile robot controller is formed by two input variables and two output variables, respectively. In Figure 30, the FIS is illustrated.

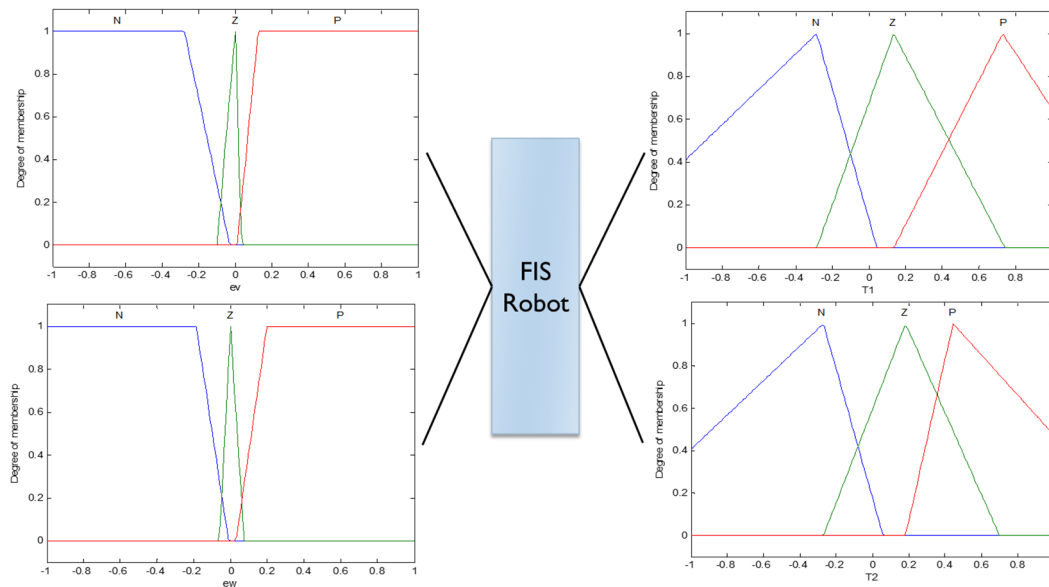


Figure 30. FIS for the Fuzzy controller.

In addition, the FIS of the mobile robot controller is formed by two input variables and two output variables, respectively. In Figure 30, the FIS is shown. The points of the input and output variables are shown in Figures 31 and 32, respectively.

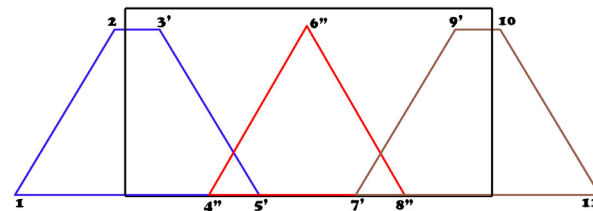


Figure 31. Input variable of the FIS in EF-FWAGWO.

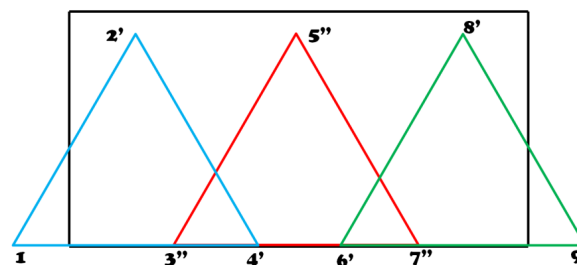


Figure 32. Output variable of the FIS in EF-FWAGWO.

In Figure 31 we are presenting the eleven points to optimizing in the input variable of the FIS.

For the output variable of the FIS was optimized eight points, which are shown in Figure 32.

In this case, to measure the performance of EF-FWA Algorithm in the control problem mentioned above, we are using the mean squared error (MSE), which is represented by the following equation:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (X_i - Y_i)^2 \quad (18)$$

Finally, Figure 33 shows the results of the simulations of the robot behavior according to the desired trajectory and the result of the best obtained MSE.

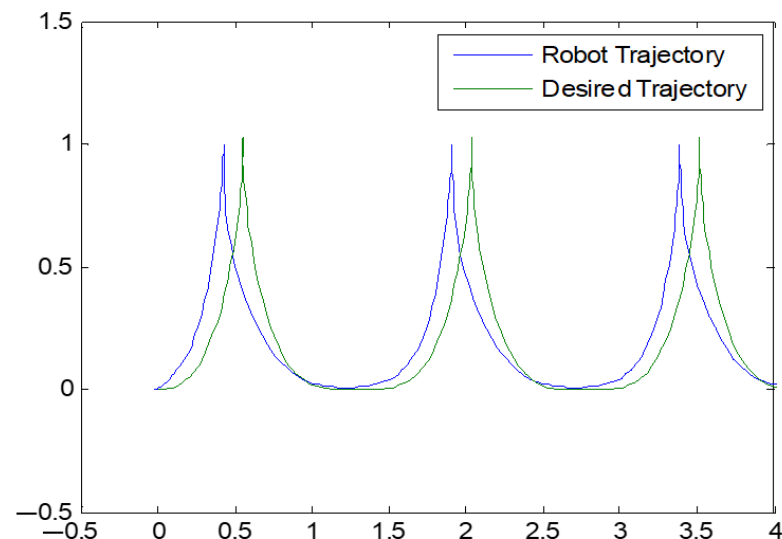


Figure 33. Best experiment obtained by EF-FWAGWO with a 0.3254 MSE.

To finish this section, it is important to mention that we can find detailed information on this mobile robot problem in the following paper [33].

5. Conclusions

In this paper, we have presented an enhancement of conventional hybridization between two great algorithms that we can find in the literature and these algorithms are the Fireworks Algorithm and the Grey Wolf Algorithm. In this work, we introduced the utilization of Fuzzy Logic in order to improve the hybridization of the two metaheuristics. We presented the Fuzzy Inference System [32], membership functions, rules, and justification of why this produces the enhancement.

In order to compare the proposed method, we presented the benchmark functions that the conventional hybridization used previously. The benchmarks are classified as unimodal and multimodal; also, we presented the results of the hypothesis tests in order to prove that Fuzzy Logic enhances the performance of the FWA-GWO. In addition, we used the same three versions that were used in the original paper, and, in this case, we noted that according to statistical tests, the versions that have better performance than the conventional ones are versions 2 and 3, respectively. In these versions, the hybridization uses the proposed method with Fuzzy Logic, which means that the enhanced version is better when the number of packs is higher, and the number of wolves is lower for each pack. This behavior is interesting because it proves that the proposed hybridization, in which the main inspiration is to divide the problem into packs, has leaders in these packs, and, finally, the rest of the solutions move according to the leaders of each pack. In addition, we showed the results with multimodal benchmarks, and we noted that the proposed method works with remarkable performance.

On the other hand, comparing methods that are similar or not similar is very complicated because the parameters or data are fuzzy. However, we have made a comparison using a similar hybridization method, and we obtained satisfactory results. The results mentioned above are presented in Tables 26–28, which show better precision.

Additionally, we applied the Fuzzy Hybrid Algorithm in a control problem; the results of the control problem using a mobile robot were satisfactory, i.e., we achieved a design of an optimal Fuzzy controller using the EF-FWAGWO algorithm with the goal of providing the control actions to the motors to minimize the error with respect to a pre-established trajectory.

In future work, we plan to consider Interval Type 2 Fuzzy Logic in the Fuzzy Hybrid Algorithm, i.e., we can use an Interval Type 2 FIS to adjust parameters in the mentioned algorithm, as this could enhance uncertainty handling in real problems. Other work could be to adjust the conventional and Fuzzy Hybrid algorithms in clustering or classification problems.

Author Contributions: Conceptualization, J.B. and L.R.; methodology, F.V. and O.C.; software, L.R.; validation, J.B. and L.R.; formal analysis, P.M. and O.C.; investigation, F.V. and P.M.; writing—original draft preparation, J.B. and L.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments: We acknowledge the support given by TecNM.

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

References

- Aladwan, F.; Alshraideh, M.; Rasol, M. A Genetic Algorithm Approach for Breaking of Simplified Data Encryption Standard. *Int. J. Secur. Its Appl.* **2015**, *9*, 295–304.
- Can, U.; Alatas, B. Physics based metaheuristic algorithms for global optimization. *Am. J. Inf. Sci. Comput. Eng.* **2015**, *1*, 94–106.
- Melián, B.; Moreno, J. Metaheurísticas: Una visión global. *Rev. Iberoam. De Intel. Artif.* **2003**, *19*, 7–28.
- Bonabeau, E.; Dorigo, M.; Theraulaz, G. *Swarm Intelligence: From Natural to Artificial Systems*; Oxford University: Oxford, UK, 1999.
- Mirjalili, S.; Mirjalili, M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [\[CrossRef\]](#)
- Tan, Y. *Fireworks Algorithm*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 355–364.
- Tan, Y.; Zhu, Y. *Fireworks Algorithm for Optimization*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 355–364.
- Zadeh, L.A. Knowledge Representation in Fuzzy Logic. *IEEE Trans. Knowl. Data Eng.* **1989**, *1*, 89–100. [\[CrossRef\]](#)
- Gavrilescu, M.; Floria, S.-A.; Leon, F.; Curteanu, S. A Hybrid Competitive Evolutionary Neural Network Optimization Algorithm for a Regression Problem in Chemical Engineering. *Mathematics* **2022**, *10*, 3581. [\[CrossRef\]](#)
- Barraza, J.; Rodríguez, L.; Castillo, O.; Melin, P.; Valdez, F. A New Hybridization Approach between the Fireworks Algorithm and Grey Wolf Optimizer Algorithm. *J. Optim. Res.* **2018**, *2018*, 6495362. [\[CrossRef\]](#)
- Abdulmajeed, N.H.; Ayob, M. A Firework Algorithm for Solving Capacitated Vehicle Routing Problem. *Int. J. Adv. Comput. Technol.* **2014**, *6*, 79–86.
- Barraza, J.; Valdez, F.; Melin, P.; Gonzalez, C. Fireworks Algorithm (FWA) with Adaptation of Parameters Using Fuzzy Logic. *Nat.-Inspired Des. Hybrid Intell. Syst.* **2017**, *667*, 313–327.
- Muro, C.; Escobedo, R.; Spector, L.; Coppinger, R. Wolf-pack (*Canis lupus*) hunting strategies emerge from simple rules. Computational simulations. *Behav. Process.* **2011**, *88*, 192–197. [\[CrossRef\]](#)
- Toolbox FWAGWO, Hybridization between the FWA and GWO Algorithms (2017). Available online: <http://www.hafsamx.org/melin/ToolFWAGWO/> (accessed on 18 November 2017).
- Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *Evolut. Comput. IEEE Trans.* **1997**, *1*, 67–82. [\[CrossRef\]](#)
- Rodríguez, L.; Castillo, O.; Soria, J. A Study of Parameters of the Grey Wolf Optimizer Algorithm for Dynamic Adaptation with Fuzzy Logic. *Nat.-Inspired Des. Hybrid Intell. Syst.* **2017**, *667*, 371–390.
- Rodríguez, L.; Castillo, O.; Soria, J.; Melin, P.; Valdez, F.; Gonzalez, C.; Martinez, G.; Soto, J. A fuzzy hierarchical operator in the grey wolf optimizer algorithm. *Appl. Soft Comput.* **2017**, *57*, 315–328. [\[CrossRef\]](#)
- Mai, D.S. Interval type-2 fuzzy logic systems optimization with swarm algorithms for data classification. In Proceedings of the 2021 13th International Conference on Knowledge and Systems Engineering (KSE), Bangkok, Thailand, 10–12 November 2021; pp. 1–5.
- Abdul-Jaleel, M.; Ali, Y.H.; Ibrahim, N.J. Fuzzy logic and Genetic Algorithm based Text Classification Twitter. In Proceedings of the 2019 2nd Scientific Conference of Computer Sciences (SCCS), Baghdad, Iraq, 27–28 March 2019; pp. 93–98.

20. Precup, R.E.; David, R.C.; Petriu, E.M. Grey wolf optimizer algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity. *IEEE Trans. Ind. Electron.* **2017**, *64*, 527–534. [\[CrossRef\]](#)
21. Simoes, M.; Bose, K.; Spiegel, J. Fuzzy Logic Based Intelligent Control of a Variable Speed Cage Machine Wind Generation System. *IEEE Trans. Power Electron.* **1997**, *12*, 87–95. [\[CrossRef\]](#)
22. Digalakis, J.; Margaritis, K. On benchmarking functions for genetic algorithms. *Int. J. Comput. Math.* **2001**, *77*, 481–506. [\[CrossRef\]](#)
23. Larson, R.; Farber, B. *Elementary Statistics Picturing the World*; Pearson Education Inc.: London, UK, 2003; pp. 428–433.
24. Molga, M.; Smutnicki, C. Test Functions for Optimization Needs. 2005, pp. 1–43. Available online: <https://robertmarks.org/Courses/ENGR5358/Papers/functions.pdf> (accessed on 25 April 2024).
25. Saremi, S.; Mirjalili, S.Z.; Mirjalili, S.M. Evolutionary population dynamics and grey wolf optimizer. *Neural Comput. Appl.* **2015**, *26*, 1257–1263. [\[CrossRef\]](#)
26. Zheng, Y.; Song, Q.; Chen, S.-Y. Multiobjective fireworks optimization for variable-rate fertilization in oil crop production. *Appl. Soft Comput.* **2013**, *13*, 4253–4263. [\[CrossRef\]](#)
27. Guillaume, S. Designing fuzzy inference systems from data: An interpretability-oriented review. *IEEE Trans. Fuzzy Syst.* **2001**, *9*, 426–443. [\[CrossRef\]](#)
28. Trojovský, P.; Dehghani, M. Hybrid leader based optimization: A new stochastic optimization algorithm for solving optimization applications. *Sci Rep* **2022**, *12*, 5549. [\[CrossRef\]](#)
29. Coma, M.; Tousi, N.M.; Pons-Prats, J.; Buggedá, G.; Bergada, J.M. A New Hybrid Optimization Method, Application to a Single Objective Active Flow Control Test Case. *Appl. Sci.* **2022**, *12*, 3894. [\[CrossRef\]](#)
30. Knypinski, Ł.; Devarapalli, R.; Gillon, F. The hybrid algorithms in constrained optimization of the permanent magnet motors. *IET Sci. Meas. Technol.* **2024**. [\[CrossRef\]](#)
31. Yu, X.; Aouari, A.; Mansour, R.F.; Su, S. A hybrid algorithm based on PSO and GA for feature selection. *J. Cybersecur.* **2021**, *3*, 117. [\[CrossRef\]](#)
32. Yue, Z.; Zhang, S.; Xiao, W. A Novel Hybrid Algorithm Based on Grey Wolf Optimizer and Fireworks Algorithm. *Sensors* **2020**, *20*, 2147. [\[CrossRef\]](#)
33. Kanayama, Y.; Kimura, Y.; Miyazaki, F.; Noguchi, T. A stable tracking control method for an autonomous mobile robot. *J. Robot. Syst.* **1997**, *14*, 697–709.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.