Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 2091337, 12 pages
https://doi.org/10.1155/2022/2091337

Research Article

Optimal Position Fuzzy Control of an Underactuated Robotic Finger

Francisco J. Espinosa Garcia,[1,2] **Esther Lugo-González**,[3] **Arturo Telléz-Velázquez**,[4] **Manuel Arias-Montiel**,[3] **and Marco Ceccarelli**[2]

[1]*Postgraduate Division, Technological University of the Mixteca, Huajuapan de León 69000, Mexico*
[2]*LARM 2, University of Rome Tor Vergata, Rome 00133, Italy*
[3]*Institute of Electronics and Mechatronics, Technological University of the Mixteca, Huajuapan de León 69000, Mexico*
[4]*Institute of Computing, Technological University of the Mixteca, Huajuapan de León 69000, Mexico*

Correspondence should be addressed to Manuel Arias-Montiel; mam@mixteco.utm.mx

Received 18 March 2022; Revised 27 May 2022; Accepted 28 June 2022; Published 20 July 2022

Academic Editor: Ardashir Mohammadzadeh

In this paper, the optimal position control of an underactuated robotic finger is presented. Two trajectories, one for the proximal and the other for the medial phalanx, are proposed in order to emulate the finger's flexion/extension movements. A Mandani fuzzy control is proposed due to the lack of a precise dynamical model of the system. In order to obtain the control parameters, an optimization strategy based on the membership functions is applied. Genetic algorithms (GA) are commonly used as an optimization method in diverse applications; however, in this case, the use of an autoadaptive differential evolution method is proposed in order to obtain a superior convergence behavior. Simulations of the virtual prototype are carried out using MATLAB/Simulink software to display the trajectory tracking. The results show that the maximum error between the proposed and obtained trajectories is $3.1352E-04$ rad.

1. Introduction

Currently, control development for robotic hands continues to be a topic of interest as researchers seek to recreate human hand interaction with their environment when developing prototypes and their interaction with the environment. This objective is difficult to achieve due to the challenging nature of controlling robotic hands, as they are relatively complex mechatronic systems, which allow the user to hold, manipulate, and make use of different objects and tools. As a possible solution, some researchers have focused on improving control tasks in order to generate a robust grasp to reliably hold any object. In addition, the selection criteria of actuators are an important factor because the size, weight, and torque are variables that must be considered. Generally, the control systems have a closed-loop structure, since they seek to reduce errors so that the hand finger can maintain a specific desired position [1, 2], by using pneumatic [3–5] and touch [6–8] sensors. In the literature, the most common control schemes are the proportional-integral-derivative (PID) control [9–11] and fuzzy control [12–14]. Controllers based on fuzzy logic are an alternative solution that does not require a mathematical model such as the PID [15]. A fuzzy logic controller (FLC) is a heuristic approach composed of a rules base proposed by the designer. The FLC is a nonlinear system with a knowledge based on fuzzy If-Then rules, and in most cases, the fuzzy rules are proposed by an expert who knows the process. In order to generate an output, membership functions are used to specify the degree of membership based on inputs. FLC must have a flexible behavior to adapt to various situations, as well as being robust to maintain the state of the desired output. The implementation of FLC is fairly common for solving problems, where (a) the systems are partially defined, (b) systems with variables that cannot be measured, and (c) system with large disturbances [12, 14, 16]. The principal fuzzy systems are Mandani and

Takagi-Sugeno. In particular, the Mandani systems use various techniques that allow fuzzy set membership function tuning, such as genetic algorithms (GA) [16], adaptive neural networks [17, 18], artificial bee colony optimization [19–21], ant colony optimization [22–24], and evolutionary algorithms [25–27].

Recently, fuzzy logic systems have been combined with nonlinear approaches in order to obtain adaptive control systems. Zhang et al. [28] proposed an FLC combined with sliding-mode technique for controlling underactuated systems. The authors provided a detailed stability analysis and a numerical example of a nonspecified system to prove the performance of the closed-loop system. However, the fuzzy system is not optimized. Li et al. [29] combined a fuzzy logic system with a back-stepping controller and a fuzzy state observer to solve the tracking problem for a class of uncertain nonlinear systems with unmodeled dynamics and disturbances. The stability of the systems is proven to be input-to-state practically stable based on the small-gain theorem, but the fuzzy logic system's membership functions are not optimized. Chang et al. [30] presented an adaptive fuzzy tracking control for a class of pure feedback and switched nonlinear systems with unknown gain. The authors did not provide details on the fuzzy system, such as the procedure to determine the membership functions.

Optimization methods have helped with the complex task of finding the appropriate parameter values and structure for the fuzzy systems. The most used algorithms for fuzzy systems optimization are GA because they are considered as global optimizers, unlike derivative methods that have the disadvantage of being prone to local minimum stagnation. They are also considered as good optimizers because of the existence of a population of potential solutions, which may be used to discard several local minima, meaning that there is a higher probability of finding the global minimum [31]. Several proposals search for optimal membership function parameters. For example, in [32] the authors proposed a methodology for optimizing a fuzzy controller for the speed of a permanent magnet synchronous motor (PMSM) without expert knowledge. For this approach, 14 variables are used for the optimization, 5 for the rule base parameters, 6 for the membership functions, and 3 for the scaling factors. A method for automatically generating fuzzy rules is presented for a system with 25 fuzzy rules needed, and each rule needs 3 bits [33]. A fuzzy controller applied to the wellhead back pressure control systems is optimized through the membership function, control rules, quantization, and scaling factors. The parameters are error, rate, scale, integral, and differential scale factors [34]. Despite this, there are other methods, such as the differential evolution (DE), which have more advantages compared to GA, one example being that they are more effective at searching for numerical problems and finding the optimum global solution [35, 36].

Motivated by the limitations of GA and other optimization methods, this paper presents the implementation of an auto adaptive differential evolution algorithm (AADE) [37] for optimizing the parameters of a fuzzy logic controller (FLC) for an underactuated robotic finger in order to track the trajectory that represents the flexion-extension finger movements. This optimization algorithm does not require gradient information, which means that the optimization problem does not need to be differentiable. In addition, AADE is based on symmetrical discourses and fuzzy normalized systems, and the parameters F (factor scale) and Cr (crossover constant) show evidence of self-adaptation during the evolution phase, unlike GA or classical differential evolution (DE) algorithms, increasing its power and execution.

The remainder of this paper is organized as follows. In Section 1, a brief description of the prototype is presented. In Section 2, the development of the FLC and the application of AADE for parameter optimization is described. In Section 3, details of the numerical implementation of the optimized FLC are provided. Final remarks and some proposed future works for this research are given in Section 4.

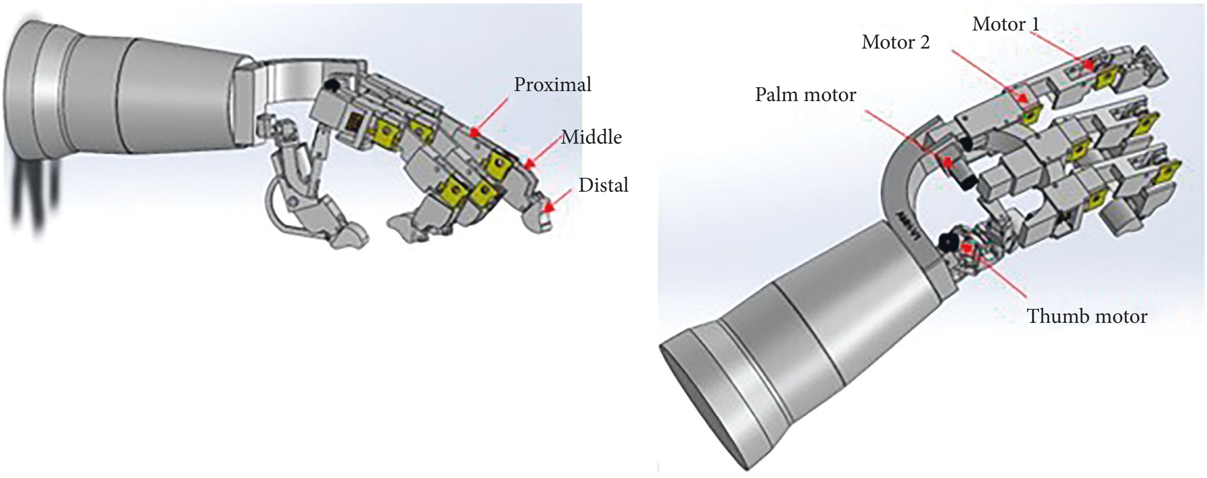
2. Materials and Methods

2.1. Prototype Design. The robotic hand AMH-II (anthropomorphic metamorphic hand), originally presented in [38], has four fingers, all of which have three joints except for the thumb. The index, middle, and ring fingers are powered by an underactuated mechanism in order to use a single motor, thereby reducing the weight of the finger. This configuration allows the hand to perform the flexion/extension finger movements. The thumb has three joints, although only two of them are used for the flexion/extension movements, while the other one is used for the rotational movement. Details on the design and kinematic analysis of the AMH-II are presented in [38]. The virtual prototype is shown in Figure 1, and as can be observed, each finger has two motors to control movement. The motors employed are Pololu micromotors 100 : 1360 RPM 6 V, and the joints are connected to these motors through a transmission system comprised of a worm-gear mechanism. The palm, unlike other robotics hands, can be moved through a motor fixed at the joint of the link, where the fingers are located and the link, where the thumb is located (thumb motor). Table 1 shows the ranges of motion for flexion/extension movements of the hand, corresponding to each finger joint (except for the thumb). Similarly, the movement measurements and ranges of the phalanges of the presented prototype are also shown.

2.2. Implemented Controller. For the control system, a Mandani fuzzy system was implemented, which includes a micro-motor that controls the position of the fingers. The transfer function corresponding to the servo motor used was obtained from [39].

$$G(s) = \frac{V(s)}{\theta(s)} = \frac{24}{s^2 + 24s + 10}, \qquad (1)$$

where $V(s)$ is the input voltage applied to the motor, while $\theta(s)$ is the rotor position in radians (rads) after applying a specific voltage. This value is used as feedback as shown in the schematic diagram in Figure 2.



(a) (b)

Figure 1: AMH-1: (a) phalanges of the hand and (b) motors' position.

Table 1: Ranges for flexion/extension movements comparison.

Phalanx	Length of the prototype (mm)	Range of human hand movement	Range of prototype movements
Distal (index and middle)	14	0–90°	0–85°
Middle (index and middle)	38.16	0–90°	0–75°
Proximal (index and middle)	41.89	0–90°	0–75°

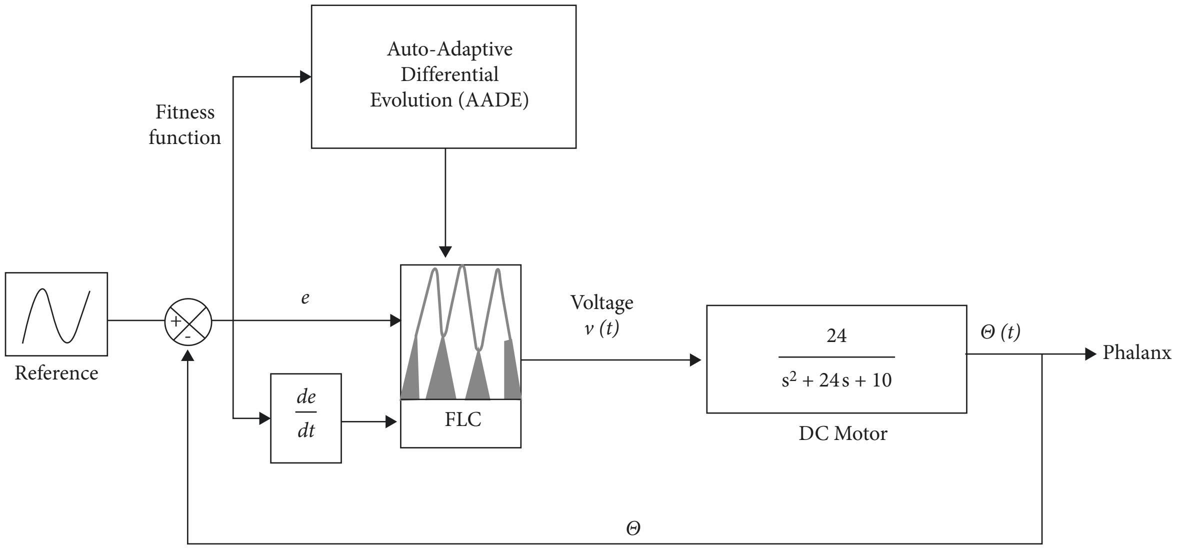


Figure 2: Block diagram of the control system.

The reference block represents the desired trajectory, while e and de are the error signal and its derivative, respectively, which serve as inputs for the fuzzy logic controller (FLC). This block sends a voltage signal to the motor block and its position is evaluated by the AADE so that the membership function parameters are changed to reduce the error.

The fuzzy system has three variables, error (e) and error change (c) are the inputs, and voltage (v) is the system output. Moreover, the variables have their own discourse universe range (Table 2) [37]. The configuration of the fuzzy system is established in 9 rules as shown in Table 3. Symbols NE, ZE, and PE are the linguistic terms for "negative error," "zero error," and "positive error"; NC, ZC, and PC are the linguistic terms for "negative change," "zero change," and

Table 2: Ranges for the fuzzy system.

Variable	Discourse ranges
Position error e	$[0, (\pi/2)]$
Error change c	$[0, (\pi/6)]$
Voltage v	$[-8.19, 8.19]$

Table 3: Rules set.

Voltage	NC	ZC	PC
Position error			
NE	NV$^{(1)}$	NV$^{(2)}$	PV$^{(3)}$
ZE	NV$^{(4)}$	ZV$^{(5)}$	PV$^{(6)}$
PE	PV$^{(7)}$	PV$^{(8)}$	PV$^{(9)}$

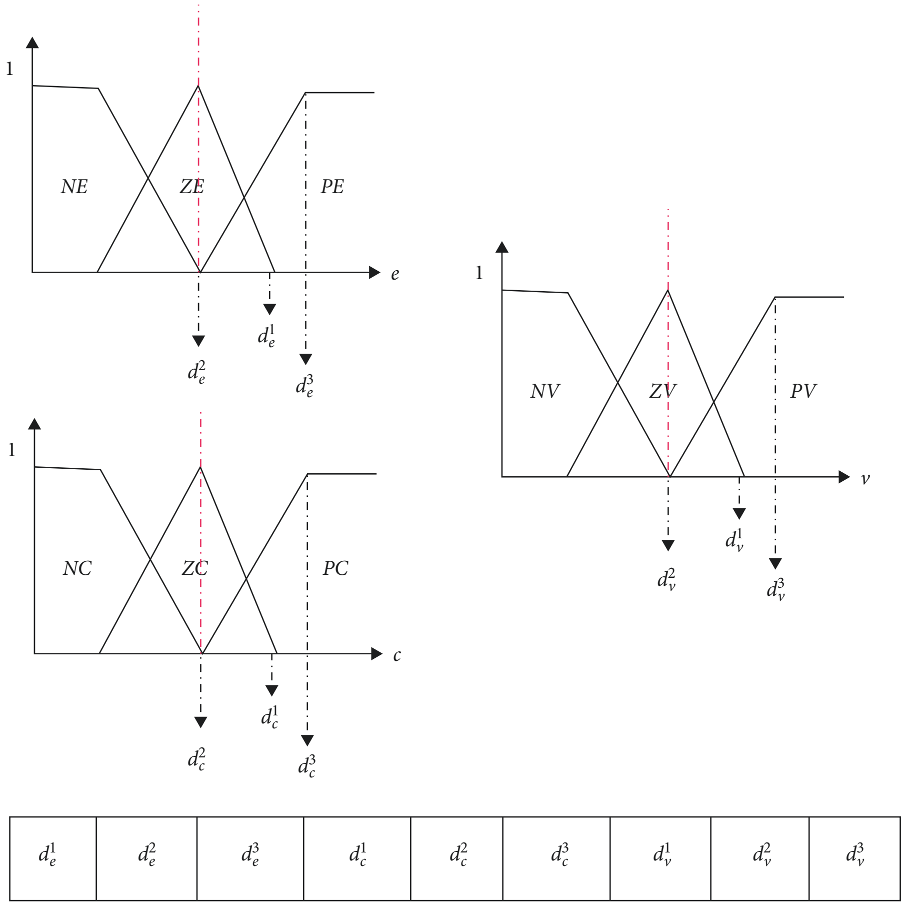


Figure 3: Example of a symmetrical variables.

“positive chance”; and NV, ZV, and PV are the linguistic terms for “negative voltage,” “zero voltage,” and “positive voltage.”

The fuzzy function proposed for the control is Gaussian for ZV, ZE, and ZC. Sigmoid functions were used, as were Z shape membership functions for NE, NC, and NV, and for the rest of the variables, the S shape function was implemented. Three parameters are needed in order to define a Gaussian function, while 2 parameters are needed for sigmoid functions. The optimization problem is then defined by 21 parameters. The number of variables requires a considerable amount of computation time and the use of resources for the optimization algorithm. Therefore, the following methodology was implemented in order to reduce the number of variables.

2.3. Methodology. In [37] a method for reducing the number of parameters for a fuzzy set control through the use of symmetrical discourses was proposed. In basic terms, these variables can distribute several sets to the positive part of the discourse universe, and then reflect them all in the negative part. It is important to consider that not all the applications can be addressed with these variables, as it is preferable that every consequent action of the fuzzy system has the same magnitude in both positive and negative directions. In this case, the symmetrical discourses can be applied because the servomotors have the same range for the output in the positive and negative response.

For example, in Figure 3, different membership functions (MFs) are shown, all of which are symmetrical. The NULL or Zero set corresponding to the triangular function is located exactly in the middle of the discourse universe (red line) while the opening of the triangular function is varied, denoted as d_e^1. In order to achieve a symmetric function, the provided value is changed to negative and as a result, the complete function is obtained. The parameters of S function are defined by d_e^2 and d_e^3, and to obtain the Z function, the S

parameter functions are reflected. With this approach, the shape of the error sets can be defined with three parameters. This same approach is also applied for the change and voltage variables. When applying this method to the proposed problem, the number of parameters is reduced from 21 to 9.

In the methodology proposed in [37], GA was used as an optimization method, while in this paper the auto adaptive differential evolution proposed by [40] is used. This is because it tends to converge to the solution in fewer iterations (ten), and does not depend on the number of the population as with GA. A description of the algorithm is presented in the next section.

2.4. *Autoadaptive Differential Evolution Algorithm (AADE).*

The AADE is a type of evolutionary algorithm used for multidimensional real-valued functions. This method does not require gradient information, which means that the optimization problem does not need to be differentiable. In addition, it has an efficient memory utilization as well as being effective and robust [36, 40–42]. The main feature of AADE is that parameters, such as F (factor scale) and Cr (crossover constant), show evidence of self-adaptation during an evolution phase, unlike GA or classical differential evolution (DE) algorithms, increasing its power and execution. Figure 4 summarizes the AADE operation flow chart. More specifically AADE's basic strategy can be described as shown below.

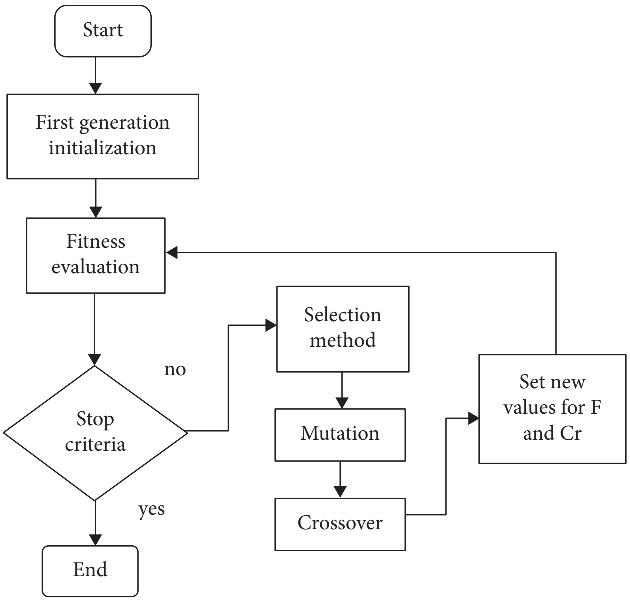


Figure 4: DE flow chart.

2.4.1. Population. The population is a vector x defined as

$$x = (x_1, x_2, \ldots, x_D), \tag{2}$$

where D is the dimensionality of the function f. The variables domains are defined by their lower and upper bounds: $x_{j,\text{low}}$ and $x_{j,\text{upp}}$; $j\,\epsilon\,\{1, \ldots, D\}$. The initial population is selected randomly between the lower ($x_{j,\text{low}}$) and upper ($x_{j,\text{upp}}$) bounds defined for each variable x_j. The bounds are specified according to the user.

2.4.2. Mutation. In every iteration (generation) G, DE uses the mutation operation for producing the donor vector. For each target vector x_i, G, $i = 1, 2, 3, \ldots$, NP, a mutant vector $v_i = \{v_{i,1}, v_{i,D}\}$ is generated according to a specific mutation scheme. The most widely used in differential evolution are

$$\begin{aligned} \frac{DE}{(\text{rand}/1v_i)} &= x_{r1} + F(x_{r2} + x_{r3}), \\ \frac{DE}{(\text{best}/1v_i)} &= x_{\text{best}} + F(x_{r1} + x_{r2}), \\ \frac{DE}{(\text{rand}/2v_i)} &= x_{r1} + F(x_{r2} + x_{r3}) + F(x_{r4} + x_{r5}), \end{aligned} \tag{3}$$

where r_1, r_2, r_3, r_4, and r_5 are random integers within the range [1, NP] and are different to index i ($r_1 \neq r_2 \neq r_3 \neq r_4 \neq r_5 \neq i$). F is a real and constant factor, which controls the amplification of the differential variation.

2.4.3. Crossover. The crossover operator is used to increase the perturbed parameter vectors' diversity. The trial vector

$$U_{i,G+1} = (U_{1i,G+1}, U_{1i,G+1}, \ldots, U_{Di,G+1}), \tag{4}$$

is formed, where

$$U_{ji,G+1} = \begin{cases} U_{ji,G+1}, & \text{if } (\text{rand}\, b(j) \le CR) \text{ or } j = rnbr(i), \\ x_{ji,G}, & \text{if } (\text{rand}\, b(j) > CR) \text{ and } j \neq rnbr(i), \end{cases} \quad j = 1, 2, \ldots, D, \tag{5}$$

where rand b (j) is the jth evaluation of a uniform random number generator within the range [0, 1], which has to be determined by the user. *rnbr* (i) is a randomly chosen index $\epsilon 1, 2, \ldots, D$, which ensures that $U_{i,G+1}$ receives at least one parameter from $U_{i,G+1}$.

2.4.4. Selection. The trial vector $U_{i,G+1}$ is compared to the target vector $x_{i,\,G}$ using the greedy criterion in order to decide whether or not it should become a member of generation $G + 1$. If vector $U_{i,G+1}$ yields a smaller cost function value than $x_{i,\,G}$, then $x_{i,G+1}$ is set to $U_{i,G+1}$; otherwise, the old value $x_{i,\,G}$ is retained.

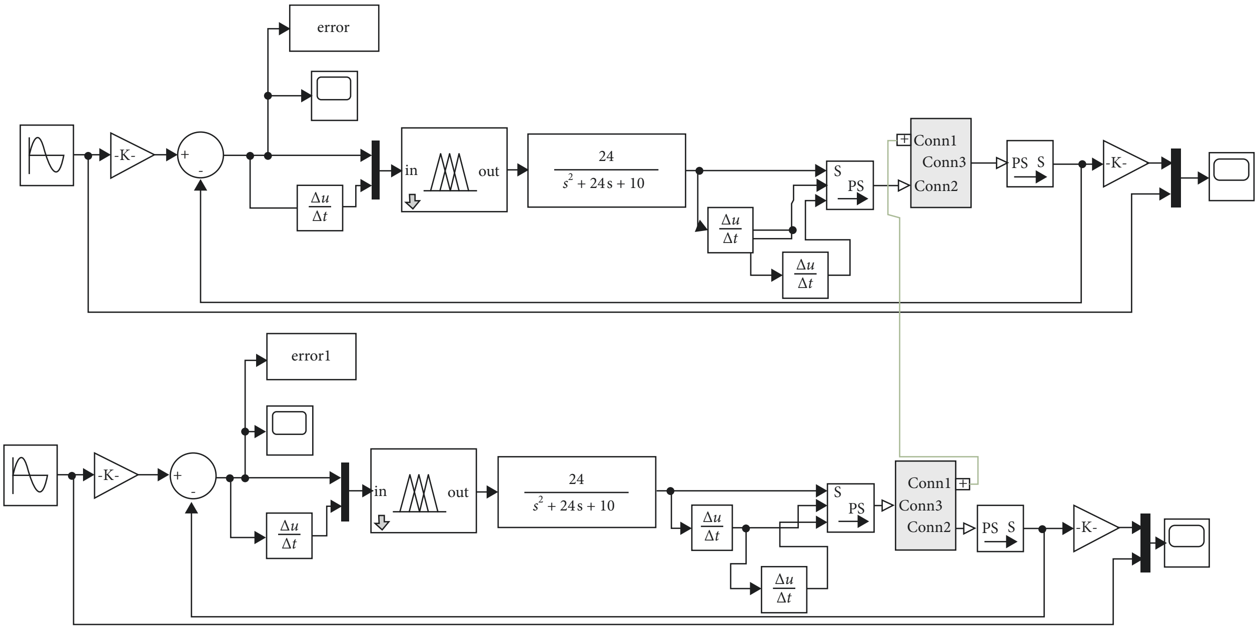


Figure 5: Block diagram of the finger control system.

2.4.5. F and Cr Value Adjust. The estimation of the *F* and Cr values depends on the following equations:

$$F_{i+1,G} = F_{i+1,G} - \mathrm{rand}_{i,j}[0,1] * \left(\frac{f\left(U_{i,G}\right) - f\left(Y_{i,G}\right)}{f\left(U_{i,G}\right)}\right),$$

$$Cr_{i+1,G} = Cr_{i+1,G} - \mathrm{rand}_{i,j}[0,1] * \left(\frac{f\left(U_{i,G}\right) - f\left(Y_{i,G}\right)}{f\left(U_{i,G}\right)}\right),$$

$$F_{G+1} = \mathrm{rand}_{i,j}[0,1] * \left(\frac{\max - \min}{\max}\right),$$

$$C_{G+1} = \mathrm{rand}_{i,j}[0,1] * \left(\frac{\max - \min}{\max}\right), \tag{6}$$

where $F_{i+1,G}$ and $\mathrm{Cr}_{i+1,G}$ represent the estimation of the *F* and Cr of each vector. $f\,(U_{i,G})$ is the new vector and $f\,(Y_{i,G})$ is the current vector. F_{G+1} and C_{G+1} are the new *F* and *CR* values for the next generation. In addition, the max and min are values obtained from an array that contains all $F_{i+1,G}$ and $\mathrm{Cr}_{i+1,G}$.

The methodology implemented is as follows:

(1) Start AADE algorithm

- (a) Set the initial parameters.
- (b) Initialize all NP individuals with random positions.
- (c) Generate three and four individuals from the current population randomly based on the mutation used.
- (d) Apply the mutation operator to form the donor vector using equation (6).
- (e) Apply the crossover operator in order to increase diversity.
- (f) Calculate the fitness of each individual.
- (g) Calculate the new *F* and *Cr* values.

(2) Once the best individual is found, every parameter must be assigned to its corresponding set.

(3) Evaluate the fuzzy system.

2.5. Simulations and Results

2.5.1. Experimental Configuration. The numerical experiments were carried out on a Legion Y540-15IRH laptop with Core i7 9th gen processor, 16 GB RAM, and a clock speed of 2.6 GHz with turbo boost of up to 4.5 GHz.

Considering the range of flexion/extension movements generated by the human finger, a middle part of a sinusoidal trajectory was proposed as a reference for the position control. Due to the robotic finger design configuration, the amplitude of the function is 1.4835 rad (85°) for the proximal phalanx and 1.3962 rad (80°) for the medial phalanx. The equation of the mean squared error (MSE) is used in order to measure the performance of the fuzzy controller. The goal function is as follows:

$$\min\left\{\frac{1}{N}\sum_{k=1}^{N}\left[x(k) - y(k)\right]^2 + M_1 h_1\right\}, \tag{7}$$

where h1 (x) = 0 ⟶ the values are in the range.h1 (x) = 1 ⟶ otherwise.

In equation (1), $x\,(k)$ represents the value at time *k*, $y\,(k)$ is the value produced by the system at time *k*, and *N* represents the total number of samples. M_1 is a constant of a very high value that penalize the goal function when the constraint fails.

The optimization problem is defined by

(i) Design variables $X = [\ d_e^1\ , d_e^2\ , d_e^3\ , d_c^1\ , d_c^2\ , d_c^3\ , d_v^1\ , d_v^2\ , d_v^3\ ]$.

Table 4: Parameters for set distribution with different mutation strategies.

Mutation strategy	$\mathbf{d}_{\mathbf{e}}^1$	$\mathbf{d}_{\mathbf{e}}^2$	$\mathbf{d}_{\mathbf{e}}^3$	$\mathbf{d}_{\mathbf{c}}^1$	$\mathbf{d}_{\mathbf{c}}^2$	$\mathbf{d}_{\mathbf{c}}^3$	$\mathbf{d}_{\mathbf{v}}^1$	$\mathbf{d}_{\mathbf{v}}^2$	$\mathbf{d}_{\mathbf{v}}^3$
DE/rand/1	0.5677	0.0873	0.0014	0.1883	0.3423	0.1870	0.9259	5.7279	7.6088
DE/best/1	0.7120	0.0957	0.0053	0.2691	0.4952	0.2781	3.6487	4.0871	6.5742
DE/rand/2	0.3860	0.0406	0.0180	0.2568	0.2537	0.3045	0.3248	6.003	6.6644

Table 5: Comparison of different mutations strategies in terms of error value.

Mutation strategy	Fitness Pha1	Fitness Pha2
DE/rand/1	$3.1352E-04$	$2.7109E-04$
DE/best/1	$7.2511E-04$	$6.4323E-04$
DE/rand/2	$4.1357E-04$	$4.0554E-04$

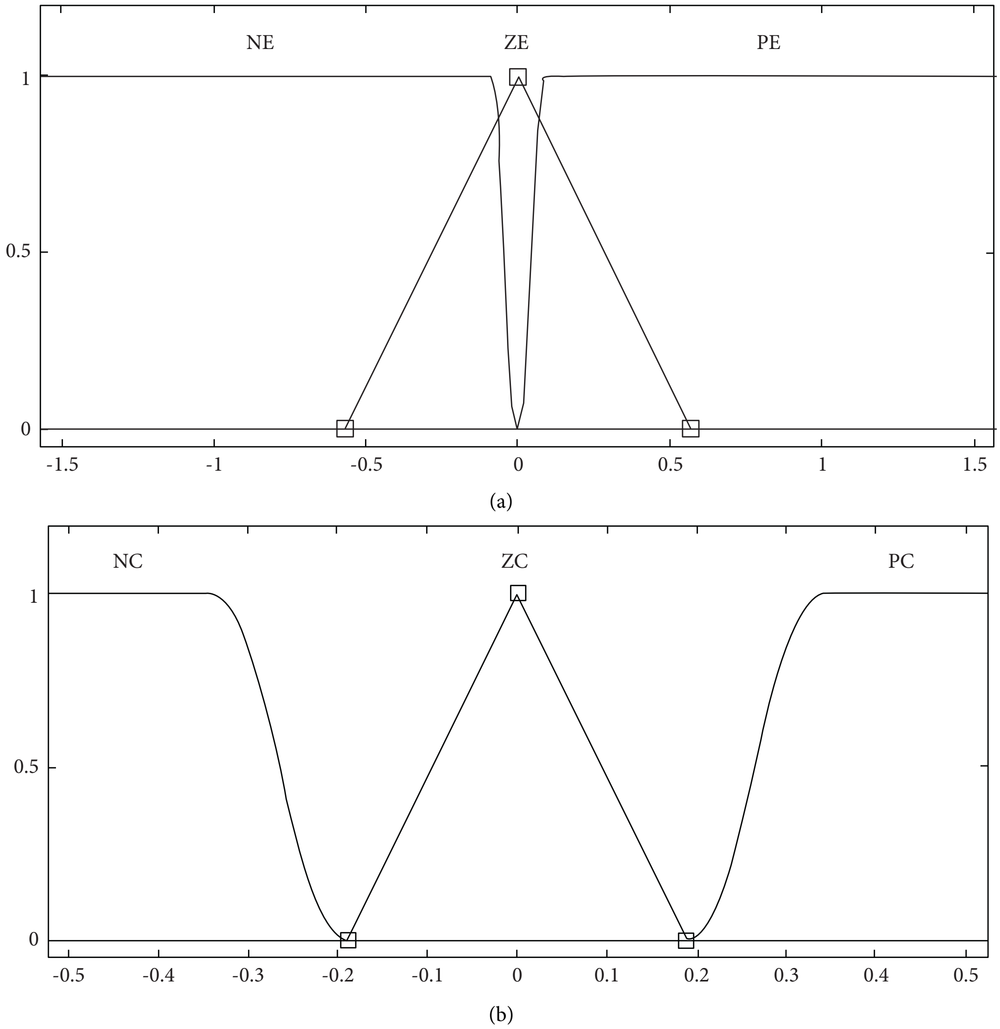


Figure 6: Membership function obtained for inputs: (a) error and (b) change.

(ii) Limits of the variables: $d_e^1, d_e^2, d_e^3 \in [0, (\pi/2)]$; $d_c^1, d_c^2, d_c^3 \in [0, (\pi/6)]$; and $d_v^1, d_v^2, d_v^3 \in [-8.16, 8.16]$.

(iii) Parameters of the algorithm:

Population size = 50.

Generation limit = 150.

$F = 0.5$.

Crossover rate = 0.2.

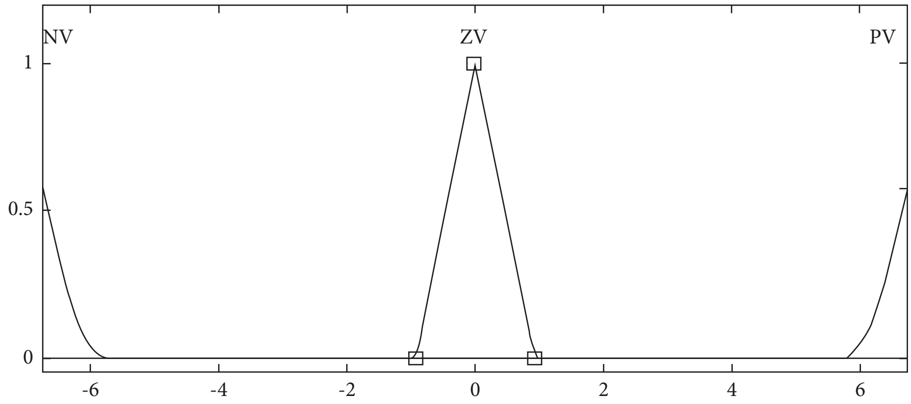


Figure 7: Membership function obtained for the output.

The proposed optimization algorithm was implemented using Matlab software, while the parallel mode was used because it increases the convergence speed, thereby demonstrating another advantage of using this approach. The algorithm was iterated using the mutation strategies from equations (5)–(7), with 30 total iterations, and 10 per each mutation strategy. In order to evaluate the proposed fuzzy control, some simulations were carried out using the Simulink environment. For simulation purposes, a finger from the hand model presented in Figure 1 was exported to Simmechanics, and the control was implemented in two parts: one for the proximal phalanx and another for the middle phalanx. This is because the distal and middle phalanxes are connected through a link resulting in a subactuated mechanism. Only one finger is simulated because it is the same movement for the other fingers. The block diagram is shown in Figure 5.

3. Results and Discussion

Table 4 shows the best performance solutions for the finger control with the three used mutation strategies.

The fitness of each solution is shown in Table 5. Pha1 and Pha2 represent the proximal and middle phalanges, respectively. All solutions are acceptable because the error is very close to zero, although the best solution is presented by DE/rand/1, followed by DE/rand/2 and DE/best/1.

Using DE/rand/1 solution, the input and output values of the fuzzy system are defined. Figures 6 and 7 show the membership functions and Figure 8 shows the 3D surface viewer of the fuzzy logic controller to get a perspective of rules design.

The results of the virtual prototype and comparison between the desired trajectory and the trajectory by the control of each phalanx are shown in Figures 9 to 11. Figure 9 shows the virtual configuration of the robotic underactuated finger, while the position of the distal phalanx depends on the movement of the middle phalanx, which is transmitted by the *P* bar.

Based on Figure 10, DTPP represents the desired trajectory of the proximal phalanx and OTPP is the obtained trajectory. The maximum rotation value of the proximal phalanx is 1.4835 rad due to the configuration of the mechanism. As can be seen in the finger's final position, the base is represented by the *B* block and the movement by the *A* block. The time simulation is 3.2 s, and the OTPP converges to DTPP with an error of $3.1352E-04$ rad. This error is more visible in two intervals, from 1.3 to 2 and from 2.5 to 3.2. Between the trajectories, there are oscillations that cannot be appreciated because they are very small ($1E-03$ rad) and appear in the interval from 0 to 1.4 s.

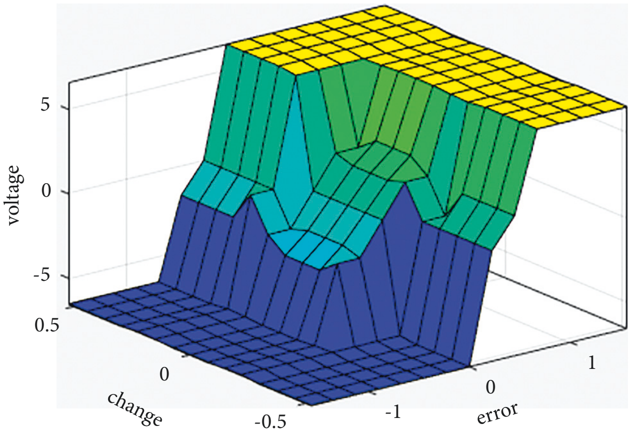


Figure 8: Rule surface.

Referring to the results of Figure 11, DTMP and OTMP are the desired and obtained trajectories of the medial phalanx. The maximum rotation value is 1.3089 rad due to the finger structure. Compared to the human finger range there is a difference of 16.66%, meaning that the finger is not able to grasp small cylindrical objects (diameter of less than 60 mm). The error between trajectories is $2.7109E-04$ rad.

According to analysis of its behavior, the controller presents a fast response due to the extremely small error tolerance (0.1 approx.), based on the membership function intersection (Figure 6(a)). Under the same conditions and modifying only the d_e^3 values, the system response speed decreases. On the other hand, after only modifying the parameter d_c^3 to values less than 0.2, the system does not present changes. It means that the motor does not change the direction of rotation and its velocity is the same under any condition. In this way, the values for d_e^1, d_e^2, and d_e^3 affect the response speed of the controller, the signal to change

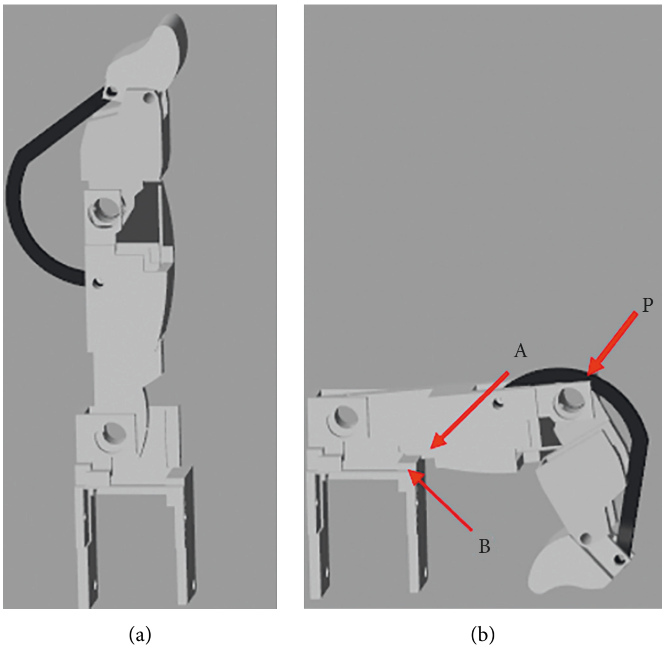


Figure 9: Finger position: (a) initial and (b) final.

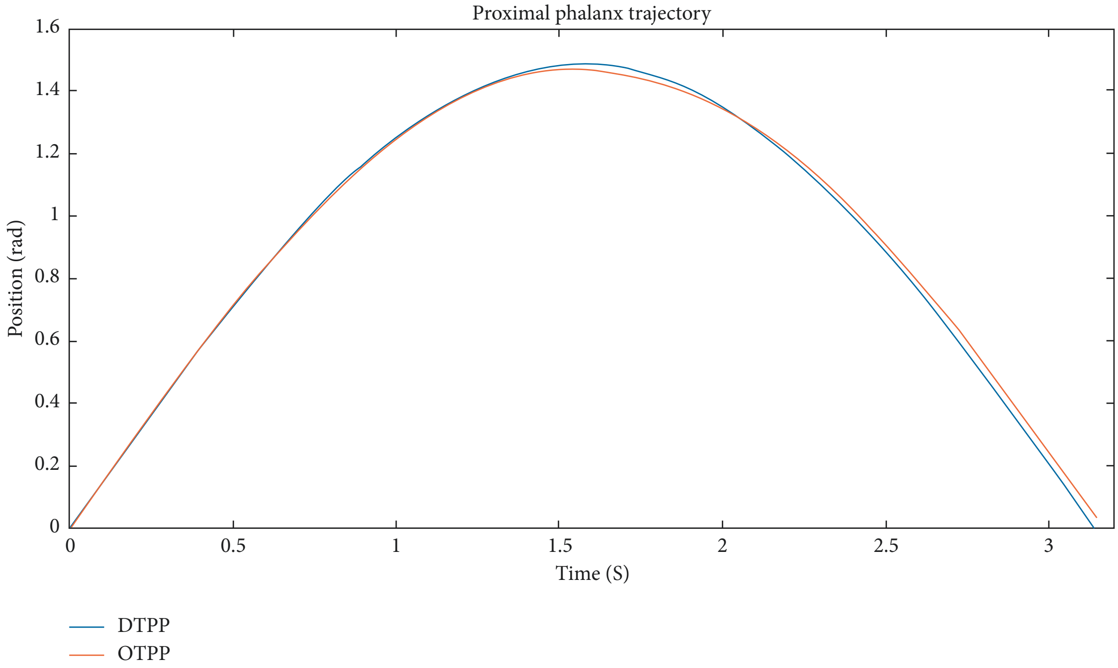


Figure 10: Comparison of desired and obtained trajectory for proximal phalanx.

rotation of the motor is defined by the parameters d_c^1, d_c^2, and d_c^3. The rest of the parameters define the motor rotation speed.

Table 6 shows a comparison between the present work and some results previously reported in the literature, and it can be observed that almost all of the finger models are underactuated. The most common technique for controlling the position of the robotic finger is the PID controller with some variants. A fuzzy controller is also used but the parameters are manually selected. Based on information in the table, in some cases, the finger dynamic model is presented in order to control the three joints. In [10, 12, 14], the authors report a step response with a steady-state error (SSE) of 0, although in [10] the time response is more than 10 sec. On the other hand, the numerical results presented in [12, 14] show settling times of less than 5 sec, but the control parameters are manually obtained. Tasar et al. [11] used a PID controller for a regulation problem of a full-actuated

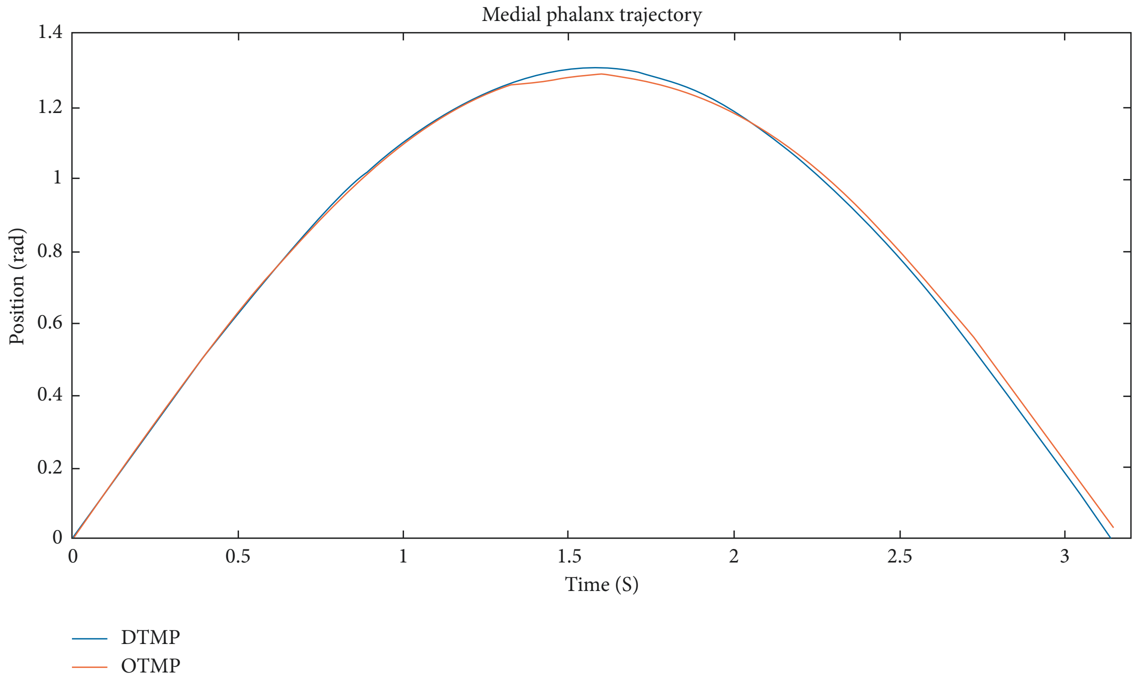


Figure 11: Comparison of desired and obtained trajectory for medial phalanx.

Table 6: Comparison between different control strategies.

Author	Control	Finger dynamic model	Trajectories	Error
Calderon et al. [10]	PI	No	Step 17 mm	SSE 0
Tasar et al. [11]	PID	Yes	Step (for three phalanxes)	SSE 0.255 deg (max)
Ghazali et al. [14]	Fuzzy-PID	No	Step 90 deg	SSE 0 deg
Ghazali et al. [14]	Fuzzy	No	Step 90 deg	SSE 0.12 deg
Raković et al. [12]	Fuzzy	No	Step 100 rad	SSE 0 rad
Jalani et al. [43]	PID	Yes	Sinusoidal [0–(−1)] rad	0.0159 rad
Jalani et al. [43]	Adaptive	Yes	Sinusoidal [0–4] rad	0.0791 rad
Jalani et al. [43]	SMC	Yes	Sinusoidal [0–(−1)] rad	0.0167 rad
Our proposal	Optimized fuzzy	No	Sinusoidal [0–1.483] rad	$3.1352E-04$ rad

robotic finger. A maximum overshoot of 14.902° was achieved for a reference of 90° for each actuated phalanx. Sinusoidal trajectories are proposed in [43] and different control techniques are presented. As in previous cases, the control parameters are tuned manually. The considered trajectory is a sinusoidal function with an amplitude range [0–(−1)] rad in PID and SMC (sliding model control) cases, which is very similar to what is reported in this work but with a larger error. The main difference with our work is that the controller parameters are automatically adjusted, and the obtained error is the smallest in all the presented trajectories.

Regarding the optimization of the fuzzy control, other works have presented methods that, in some cases, optimize more than 10 variables [29]. Moreover, the most common algorithm is GA, although one of the main problems with this type of algorithm is that it is largely dependent on the population size. Moreover, the selection of the initial parameters represents a challenge because selecting the wrong values can lead to the algorithm not converging. In order to avoid this problem, the methods applied in this paper show that it is possible to reduce the number of parameters, considering that the fuzzy set is symmetrical. Instead of using a GA algorithm, it was decided to implement an AADE algorithm because the control parameters F and CR are automatic and the time convergence is smaller compared to GA. Another advantage is that the AADE has the potential to generate a fuzzy control with no previous knowledge of the system.

4. Conclusion

When removing the empiricism of a fuzzy controller, a nonlinear problem with several dimensions appears, which can be solved by focusing on the membership function to generate a desired control. In this paper, the auto adaptive differential evolution algorithm was implemented because of the limitations of other methods such as genetic algorithms, artificial bee colonies, or ant colonies. The chosen method performs a sophisticated local search through the recombination method, and the way in which the mutated method vector is calculated means that the algorithm depends less on an initial position. Furthermore, another advantage is that F and Cr values are calculated automatically. Additionally, the use of the symmetrical discourses is different to other methods for finding the membership function mentioned in the introduction. Specific information about the system is neither not needed nor necessary to work with all the variables shown in the proposed problem. At the start of the process, it was composed of 21 variables, and before applying the method, the variables were reduced to 9. With the obtained vector, the MFs were generated to the fuzzy control. From the simulation, the obtained results proved that the control is able to perform flexion-extension movements following the proposed trajectory with an error of almost zero, despite not having a mathematical model of the robotic hand. It would be very difficult to achieve these results by selecting the controller parameters empirically.

The following step in this research is to prove experimentally the effectiveness of the proposed controller by carrying out tests on the physical prototype reported in [38]. In addition, future studies will focus on the optimization of membership functions to generate movement in the palm and doing fingers able to perform different kinds of grasps to evaluate the efficiency of this type of control scheme. Another interesting topic to be addressed in future works is the consideration of type-3 fuzzy logic systems (T3FLS) to improve the control system performance. T3FLS have been proposed to handle an increased number of uncertainties [44]. However, their main disadvantages are that the structure of the fuzzy sets has been assumed to be constant and the membership computation is complicated. Moreover, studies on the parameter optimization of T3FLS have been scarcely reported in literature [45]. Therefore, combining AADE algorithms with T3FLS could be a promising alternative to provide robustness and improve the fuzzy logic controller performance.

Data Availability

The data that support the findings of this study are available upon request to the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors are grateful to the National Council of Science and Technology (CONACYT) and to the Project 621 from the Cathedra's Program for supporting the research presented in this paper.

References

[1] Y. Wang, X. Wu, D. Mei, L. Zhu, and J. Chen, "Flexible tactile sensor array for distributed tactile sensing and slip detection in robotic hand grasping," *Sensors and Actuators A: Physical*, vol. 297, p. 111512, 2019.

[2] S. Bilgin, Y. Üser, and M. Mercan, "Robotic hand controlling based on flexible sensor," *International Journal of Engineering and Applied Sciences*, vol. 8, no. 4, p. 49, 2016.

[3] H. Liu and D. Geng, "Kinematics modeling and grasping experiment of pneumatic four-finger flexible robotic hand," *Proceedings of the Institution of Mechanical Engineers - Part C: Journal of Mechanical Engineering Science*, vol. 235, no. 24, pp. 7894–7906, 2021.

[4] P. M. Khin, J. H. Low, M. H. Ang, and C. H. Yeow, "Development and grasp stability estimation of sensorized soft robotic hand," *Frontiers in Robotics and AI*, vol. 8, p. 619390, 2021.

[5] P. Svensson, C. Antfolk, N. Malesevic, and F. Sebelius, "Characterization of pneumatic touch sensors for a prosthetic hand," *IEEE Sensors Journal*, vol. 20, no. 16, pp. 9518–9527, 2020.

[6] Z. Deng, Y. Jonetzko, L. Zhang, and J. Zhang, "Grasping force control of multi-fingered robotic hands through tactile sensing for object stabilization," *Sensors*, vol. 20, no. 4, p. 1050, 2020.

[7] A. M. Almassri, W. Z. Wan Hasan, S. A. Ahmad et al., "Pressure sensor: state of the art, design, and application for robotic hand," *Journal of Sensors*, vol. 2015, pp. 1–12, 2015.

[8] A. Saudabayed and H. A. Varol, "Sensors for robotic hands: a survey of the state of the art," *IEEE Access*, vol. 3, pp. 1765–1782, 2015.

[9] M. K. Kasim, R. L. Shauri, and K. Nasir, "PID position control of three-fingered hand for different grasping styles," in *Proceedings of the 6th Control and System Graduate Research Colloquium (ICSGRC)*, pp. 7–10, Shah Alam, Malaysia, August 2015.

[10] C. A. Calderon, C. Ramirez, V. Barros, and G. Punin, "Design and deployment of grasp control system applied to robotic hand prosthesis," *IEEE Latin America Transactions*, vol. 15, no. 2, pp. 181–188, 2017.

[11] B. Tasar, A. B. Tatar, A. K. Tanyildizi, and O. Yakut, "Design, dynamic modelling and control of wearable finger orthosis," *Journal of Mechanics in Medicine and Biology*, vol. 21, no. 1, 2021.

[12] M. Raković, G. Anil, Z. Mihajlović et al., "Fuzzy position-velocity control of underactuated finger of FTN robot hand," *Journal of Intelligent and Fuzzy Systems*, vol. 34, no. 4, pp. 2723–2736, 2018.

[13] X. h. Liu, X. h. Chen, X. h. Zheng, S. p. Li, and Z. b. Wang, "Development of a GA-fuzzy-immune PID controller with incomplete derivation for robot dexterous hand," *The Scientific World Journal*, vol. 2014, pp. 1–10, 2014.

[14] R. Ghazali, M. Z. Saad, S. Y. S. Hussien, M. H. Jali, F. N. Zohedi, and T. A. Izzuddin, "Intelligent controller design for multifunctional prosthetics hand," *International Journal of Mechanical Engineering and Robotics Research*, vol. 6, no. 6, pp. 495–501, 2017.

[15] K. Sato and G. J. Maeda, "A practical control method for precision motion-Improvement of NCTF control method for continuous motion control," *Precision Engineering*, vol. 33, no. 2, pp. 175–186, 2009.

[16] I. Yusuf, Y. Yusuf, and N. Iksan, "FGA temperature control for incubating egg," *Advances in Fuzzy Systems*, vol. 2012, no. 1, pp. 1–8, 2012.

[17] W. S. Chan, C. Y. Lee, C. W. Chang, and Y. H. Chang, "Interval type-2 fuzzy neural network for ball and beam systems," in *Proceedings of the International Conference on Systems Science and Engineering*, pp. 315–320, Taipei, Taiwan, July 2010.

[18] Q. Zhu, Y. Han, C. Cai, and Y. Xiao, "Robust optimal navigation using nonlinear model predictive control method combined with recurrent fuzzy neural network," *Mathematical Problems in Engineering*, vol. 2018, pp. 1–19, 2018.

[19] L. Amador-Angulo and O. Castillo, "Optimization of the type-1 and type-2 fuzzy controller design for the water tank using the bee colony optimization," in *Proceedings of the Conference on Norbert Wiener in the 21st Century*, pp. 1–8, Boston, USA, June 2014.

[20] M. Nikolić, M. Šelmić, D. Macura, and J. Ćalić, "Bee colony optimization metaheuristic for fuzzy membership functions tuning," *Expert Systems with Applications*, vol. 158, p. 113601, 2020.

[21] M. S. Jacob and P. Selvi Rajendran, "Fuzzy artificial bee colony-based CNN-LSTM and semantic feature for fake product review classification," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 1, 2022.

[22] A. Rezoug, Z. Achour, and M. Hamerlain, "Ant colony optimization of type-2 fuzzy helicopter controller," in *Proceedings of the International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1548–1553, Bali, Indonesia, December 2014.

[23] A. Ragmani, A. Elomri, N. Abghour, K. Moussaid, and M. Rida, "An improved hybrid fuzzy-ant colony algorithm applied to load balancing in cloud computing environment," *Procedia Computer Science*, vol. 151, pp. 519–526, 2019.

[24] V. Sangeetha, R. Krishankumar, K. S. Ravichandran et al., "A fuzzy gain-based dynamic ant colony optimization for path planning in dynamic environments," *Symmetry*, vol. 13, no. 2, p. 280, 2021.

[25] L. D. Seixas and F. C. Correa, "Comparison between genetic algorithm and differential evolution tuning a fuzzy controller," *Mathematics in Engineering Science and Aerospace (MESA)*, vol. 11, no. 3, pp. 1–11, 2020.

[26] S. Tsafarakis, K. Zervoudakis, A. Andronikidis, and E. Altsitsiadis, "Fuzzy self-tuning differential evolution for optimal product line design," *European Journal of Operational Research*, vol. 287, no. 3, pp. 1161–1169, 2020.

[27] O. Castillo, F. Valdez, J. Soria et al., "Optimal design of fuzzy systems using differential evolution and harmony search algorithms with dynamic parameter adaptation," *Applied Sciences*, vol. 10, no. 18, p. 6146, 2020.

[28] H. Zhang, N. Xu, G. Zong, and A. F. Alkhateeb, "Adaptive fuzzy hierarchical sliding mode control of uncertain underactuated switched nonlinear systems with actuator faults," *International Journal of Systems Science*, vol. 52, no. 8, pp. 1499–1514, 2021.

[29] Y. Li, N. Xu, B. Niu, Y. Chang, J. Zhao, and X. Zhao, "Small-gain technique-based adaptive fuzzy command filtered control for uncertain nonlinear systems with unmodeled dynamics and disturbances," *International Journal of Adaptive Control and Signal Processing*, vol. 35, no. 9, pp. 1664–1684, 2021.

[30] Y. Chang, P. Zhou, B. Niu et al., "Switched-observer-based adaptive output-feedback control design with unknown gain for pure-feedback switched nonlinear systems via average dwell time," *International Journal of Systems Science*, vol. 52, no. 9, pp. 1–15, 2021.

[31] O. Castillo and P. Melin, "A review on the design and optimization of interval type-2 fuzzy controllers," *Applied Soft Computing*, vol. 12, no. 4, pp. 1267–1278, 2012.

[32] N. Öztürk and E. Çelik, "Speed control of permanent magnet synchronous motors using fuzzy controller based on genetic algorithms," *International Journal of Electrical Power & Energy Systems*, vol. 43, no. 1, pp. 889–898, 2012.

[33] H. Zhang, B. Zhang, and F. Wang, "Automatic fuzzy rules generation using fuzzy genetic algorithm," in *Proceedings of the Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 107–112, Tianjin, China, August 2009.

[34] H. Liang, J. Zou, K. Zuo, and M. J. Khan, "An improved genetic algorithm optimization fuzzy controller applied to the wellhead back pressure control system," *Mechanical Systems and Signal Processing*, vol. 142, p. 106708, 2020.

[35] N. Lukman, M. Irfan, A. Nugraha, and J. Jumadi, "Comparison between genetic algorithm with differential evolution in study scheduling," in *Proceedings of the 5th Annual Applied Science and Engineering Conference (AASEC 2020)*, Bandung, Indonesia, April 2020.

[36] M. Georgioudakis and V. Plevris, "A comparative study of differential evolution variants in constrained structural optimization," *Frontiers in Built Environment*, vol. 6, 2020.

[37] A. Téllez-Velázquez, H. Molina-Lozano, L. A. Villa-Vargas et al., "A feasible genetic optimization strategy for parametric interval type-2 fuzzy logic systems," *International Journal of Fuzzy Systems*, vol. 20, no. 1, pp. 318–338, 2018.

[38] F. J. Espinosa-Garcia, R. Tapia-Herrera, E. Lugo-Gonzalez, and M. Arias-Montiel, "Development of a robotic hand based on a metamorphic mechanism for extending the thumb's functionality," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 43, pp. 1–16, 2021.

[39] B. Pillai and K. T. Nair, "Intelligent adaptive control for DC servo motor position control in LabVIEW," in *Proceedings of the International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICCT)*, pp. 981–985, Kerala, India, July 2017.

[40] V. Sharma, S. Agarwal, and P. K. Verma, "Auto adaptive differential evolution algorithm," in *Proceedings of the Third International Conference on Computing Methodologies and Communications (ICCMC)*, pp. 958–962, Erode, India, March 2019.

[41] D. Zaharie, "A comparative analysis of crossover variants in differential evolution," in *Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT)*, pp. 171–181, Wisla, Poland, October 2007.

[42] A. Slowik and H. Kwasnicka, "Evolutionary algorithms and their applications to engineering problems," *Neural Computing & Applications*, vol. 32, no. 16, pp. 12363–12379, 2020.

[43] J. Jalani, G. Herrmann, and C. Melhuish, "Robust trajectory following for underactuated robot fingers," in *Proceedings of the UKACC International Conference on Control*, pp. 1–6, Coventry, UK, September 2010.

[44] A. Mohammadzadeh, M. H. Sabzalian, and W. Zhang, "An interval type-3 fuzzy system and a new online fractional-order learning algorithm: theory and practice," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 9, pp. 1940–1950, 2020.

[45] S. N. Qasem, A. Ahmadian, A. Mohammadzadeh, S. Rathinasamy, and B. Pahlevanzadeh, "A type-3 logic fuzzy system: optimized by a correntropy based Kalman filter with adaptive fuzzy kernel size," *Information Sciences*, vol. 572, pp. 424–443, 2021.