Hindawi
Complexity
Volume 2022, Article ID 4284378, 17 pages
https://doi.org/10.1155/2022/4284378

Research Article

Filling Control of a Conical Tank Using a Compact Neuro-Fuzzy Adaptive Control System

Helbert Espitia-Cuchango,[1] **Iván Machón-González**,[2] **and Hilario López-García**[2]

[1]*Facultad de Ingeniería, Universidad Distrital Francisco José de Caldas, Bogotá 110231, Colombia*
[2]*Departamento de Ingeniería Eléctrica, Electrónica de Comunicaciones y de Sistemas, Universidad de Oviedo, Campus de Viesques, Gijón 33204, Spain*

Correspondence should be addressed to Helbert Espitia-Cuchango; heespitiac@udistrital.edu.co

Received 1 July 2022; Revised 3 October 2022; Accepted 7 October 2022; Published 29 October 2022

Academic Editor: Guang Li

This document describes the implementation of a conical tank control system using an adaptive neurofuzzy system. For implementation, an indirect approach is used where the controller is optimized using the model obtained during the plant identification carried out using data obtained during the system operation. Furthermore, implementation includes training of neuro fuzzy-systems and application to control a conical tank. Regarding plant identification, preliminary training takes place using data obtained for different input values. The controller configuration is established considering the analogy with a discrete-time linear system. The simulation shows that the control system manages to approach the desired response given by the considered reference model.

1. Introduction

Adaptive control is framed in the field of control systems to face uncertainties. According to [1], adaptive control systems are suitable for monitoring performance using varied and unknown parameters [1]. The main difference between adaptive and linear controllers lies in the adjustment capacity of the controller to manage unknown uncertainties in the model [2].

With regard to developments of adaptive control, controller technique designs are proposed to manage nonlinear and time-varying uncertainties. Such developments can cover larger systems with higher nonlinear uncertainties. Adaptive control has real-world actual applications such as developments in nonlinear systems with variable control gain in time, as observed in [3].

Simultaneously, adaptive dynamic programming (ADP) has proven its efficiency as an effective method for the optimum control of nonlinear systems. Nevertheless, since the structure of the ADP requires a control input to satisfy the condition of initial permissible control, then control performance may be deteriorated due to abrupt parameter changes or a failure in the system. According to [4], for this reason, a multiple model adaptive control (MMAC) is proposed employing multiple ADP models where combined sub-controllers are run in parallel offering multiple initial conditions in different settings, including the configuration of a commutation rate to determine suitable initial conditions for the current system.

As stated in [5], parameter convergence in the adaptive control produces improvements in the system performance, including accurate online identification, exponential monitoring, and robust adaptation without parameter deviation. Nevertheless, parameter convergence must fulfill the strong persistent-excitation (PE) or sufficient-excitement (SE) as a guarantee of parameter convergence in the classic adaptive control.

According to [6], techniques based on gradient calculations offer efficient and practical methods to adjust the parameters in the adaptive control system. An alternative to enhance the performance of the adaptive control system employing gradient-based algorithms consists of an adequate preliminary configuration of the systems used for plant identification, and the controller. In this regard, a fuzzy system allows the establishment of a structure and a system preliminary configuration employed to identify the plant and the controller optimization [7]. Additionally, only data training is required when implementing the adaptive control system for the plant model adjustment, which is useful in highly complex systems with uncertainty and variance [1].

1.1. Adaptive Control Systems. Regarding adaptive control application developments, [8] presents an extension of model reference adaptive control (MRAC) to systems based on the fractional calculus theory employed for adaptive control. The authors designed two laws: one of control and one of incommensurable fractional adaptation for both the fractional plant and the fractional reference model. The stability and convergence are analyzed including the fractional integrator model in frequency and the theory of Lyapunov.

Meanwhile, [9] proposes a scheme for nonlinear plants that gain variable control in time and plant coefficients with variability in time, which requires a Brunovsky-type plant model with approximating polynomials. The authors added a scheme of robust control to the plant. A combination of entries is used to achieve sturdiness with dead zone updating laws.

Considering [10], the cabin pressure control system (CPCS) is an essential part that guarantees the aircraft structure and the crew's safety; however, the CPCS shows potential flaws in sensors and actuators. Consequently, authors in [10] suggest a reconfiguration method based on simple adaptive control in compensation for these adverse effects.

According to [11], nonlinear systems can be modeled as linear systems in parts of multiple operative points. Each operative point is modeled as a switch among constituent linear systems. Regarding [11], an adaptive linear switch controller by parts is proposed to enhance the response time and tracking performance of a control system for a hydraulic actuator, which is essentially linear by parts. The controller, which consists of proportional, integral, and derivative controllers (PID) and MRAC chooses adaptatively the proportion of both components to achieve faster time response and improved tracking performance.

Related work with adaptive control is seen in [12] where the objective is the proper diagnostic and flaw estimations in an automotive magnetorheological damper; in addition, the authors suggest a robust linear parameter-varying (LPV) estimate the lack of power caused by a leak in the damper of one side of the vehicle. An adaptive system of vibration control or adaptive vibration control system (AVCS) is also suggested after identifying the faulty damper to diminish the effect of the failure by using compensation forces from the operational dampers.

1.2. Control Systems with Neural Networks. In terms of control development systems employing neural networks, [13] proposes a control strategy of nonlinear systems with unknown dynamics through a set of local linear models from a gas neural network under supervision. The direct model of the plant comprises a linear approximation by parts of the nonlinear system, and each neuron represents a local linear model for which a linear controller is designed. The neural gas model works simultaneously as an observer and controller. A control of feedback status is used through the estimation of status variables of the local transfer function emanating from the local linear model.

According to [14], the control of a ship's rudder during a mission on the sea reveals complexity. In this regard, [14] proposes a quantum neural network (QNN) to make use of the learning capacity and learning speed to work as the feedback control hierarchy model for planning strategy and intelligent control of the ship.

Another work is observed in [15] considering the advantages of a radial basis function (RBF) and a traditional PID which is suggested as a controller based on the supervision control method of neural network RBF (PID-RBF); this method performs the adaptive adjustment of the stable tracking signal of the system. The authors also display a PID controller rooted in the supervision control strategy of neural networks; this control strategy adopts the supervision control method feed-forward and feedback.

Meanwhile, in [16] the design of an adaptive switch controller (ASC) for a multiple input multiple output (MIMO) system is proposed. The suggested method performs the change online between the neural adaptive PID controller (APID) and the neural indirect adaptive controller (IAC). Considering the scheme IAC based on a neural network, the law of adaptation is established by the method of gradient descent (GD). The adaptive controller PID is built based on the neural network that combines the PID control and the explicit neural structure. The training strategy consists of the adjustment of the neural controller line weights employing the backpropagation algorithm to choose the suitable PID gains so that the error between the reference signal and the actual output of the system converges to zero.

In reference to other adaptive controls with neural network approaches, in [17] an adaptive dynamic surface control scheme based on a neural network is proposed, and a transformed function of error tracking for a system of control excitation of the generator equipped with a static VAR compensator (SVC) and unknown parameters. The predetermined efficiency of the error tracking is guaranteed by combining the transformed function of the error tracking.

About recent and noteworthy developments, a neural network-based adaptive control approach for stabilizing the air gap in a non-linear maglev vehicle is proposed by [18]. The system is designed considering the asymptotic stability associated with the control law. The controller includes a radial neural network connected inside the controller to deal with uncertainty. Lyapunov stability analysis is used to demonstrate the stability of the maglev control system. Simulation outcomes show that the control scheme obtains more suitable levitation performance, achieving accurate estimation of disturbances.

1.3. Fuzzy Adaptive Control Systems. Regarding adaptive control applications with fuzzy logic, in [19] a fuzzy adaptive controller for robotic manipulators is designed, considering external disturbances and modeling errors. The dynamics of the robotic manipulator refers to multiple input and multiple output system. A scheme of adaptive fuzzy logic control is studied employing the sliding control theory, which adopts adaptive fuzzy logic systems to assess uncertainties employing a filtered error to compensate for the approximation errors. By employing Lyapunov's stability theory, it is demonstrated that errors asymptotically converge to zero.

According to [20], for the design of controllers for wheeled mobile robots (WMR) the exact position of the mass center is challenging to measure since WMR is an uncertain non-linear typical system. In this regard, in [20] an adaptive fuzzy control scheme for tracking pathways (routes) for mobile robots is proposed. The fuzzy system is used to approximate the unknown pathways, and a robust controller is built to compensate for the approximation error.

On the other hand, [21] presents an adaptive controller for chaotic systems MIMO with system uncertainties. The matrix decomposition theory is utilized by decomposing the control gain matrix into a positive matrix, a diagonal matrix, and a unitary upper triangular matrix. Moreover, an adaptation law proposed called proportional integral (PI) is used for parameter update.

In [22] a type 2 fuzzy controller Takagi Sugeno (TS) for a non-linear system is proposed, which combines different types of supervised control as direct, indirect, and compensation to build the controller. Employing the synthesis method of Lyapunov the global stability and the close loop system convergence are analyzed with the condition that all the variables are evenly delimited, besides the adaptive laws of the system parameters are also given.

Meanwhile, in [23] an adaptive fuzzy control of variable structure PID is proposed for tracking the position of a permanent magnet synchronous motor (PMSM) employed on an electric extremity exoskeleton robot (EEER). A variable structure of a sliding mode control (SMC) is used regarding the traditional PID. The variable structure is designed according to the surface of the slider mode given by the system state equation. Regarding the system vibration of the slider mode, the fuzzy inference mode is adopted to adjust the PID parameters in an adaptive fashion in real-time to attenuate the vibration and to enhance control accuracy and the dynamic performance of the system. The use of the Lyapunov analysis allows us to demonstrate that this algorithm converges into the sliding surface and guarantees system stability.

Regarding relevant developments in neural-fuzzy control, [24] is presented a maglev train system with an adaptive neural-fuzzy robust position control architecture. The proposal also discusses the design and execution of the magnetic suspension controller, including the construction of a mathematical model system with magnetic suspension. The authors carry out the proportional integral derivative (PID) controller analysis to demonstrate its sensitivity to disturbances. In addition, a sliding mode controller is introduced as a means to reject parameter perturbations and disturbances by employing an adaptive fuzzy approximator, the neural-fuzzy switching law, and sliding mode control. The experimental results evidence that the robust controller noticeably decreases parameter perturbations and disturbance.

1.4. Article Focus and Document Organization. Regarding previous works that support the development shown in this paper, [25] is described the design methodology for fuzzy inference systems based on boolean relations; in this order, the proposed adaptive control system is rooted in the use of compact fuzzy systems based on boolean relations described in [26], where neuro-fuzzy architectures I and II are proposed for identification and control.

In previous implementations using fuzzy systems-based boolean relations, in [27] the adaptive control for a power distribution system is implemented, and in [28] the control of a MIMO system is performed. In the training of the neuro-fuzzy controller, in [29] a system to control an automatic voltage regulator is designed and optimized by employing architecture II for the controller. Meanwhile, reference [30] displays the optimization of a PI controller based on architecture II. It is noteworthy that works in references [29, 30] consider linear models of plants.

In this work, for the implementation of the control system, architecture I is used for identification, and architecture II for control. The deduction of the training equations and the simulation of the system are presented, verifying the performance of the adaptive control system.

The configuration of the controller with architecture II is achieved using fuzzy sets to model positive and negative values. On the other hand, for the identification with architecture I, a preliminary identification is made with data obtained from the plant for different input values, since the assignment of the fuzzy sets in architecture I is carried out by distributing the fuzzy sets in the universes of discourse associated with the inputs.

According to [31] adaptive control problems, it is sought to determine stabilization schemes that counter the effect of uncertain parameters in a robustness perspective. In this order, the scheme proposed displays an alternative for adaptive control using fuzzy compact systems that describe a direct logical relationship between inputs to outputs. The proposed adaptive control system employs a fuzzy compact system for identification, and another for the controller, in a way that the identification and controller training is performed iteratively.

The document is organized as follows. Section 2 describes the adaptive control system employed, then Section 3 shows the architecture of the neuro-fuzzy system used for plant identification and the respective training equations. Subsequently, Section 4 describes the architecture of the neuro-fuzzy controller and the respective deduction of the equations for the optimization of the controller. In Section 5, the dynamic model of the conical tank is described, with the experimental results shown in Section 6. Finally, the discussion and conclusions of the work are presented in Sections 7 and 8.

2. Adaptive Neuro-Fuzzy Control System

Both adaptive control and robust control are techniques employed when the mathematical model is incapable of representing the actual system accurately. The purpose of robust control is to determine a law of control to maintain the system response and the error signal within the limits predetermined when having uncertainty in the model. As a goal, adaptive control searches the adaptability of the closed-loop system to variant behavioral circumstances in the system dynamics and disturbances [32]. In addition, adaptive control allows the observation of two dynamic behaviors in constant evolution with different time lapses. Consequently, parameter changes can be observed at a slow scale and thus the rate of adjustment of the parameters. Likewise, on a fast scale, the closed-loop dynamics in the system are observed [32].

The adaptive control technique is capable of facing disturbances, uncertainties, and variances of operative conditions in the system dynamic employing an adaptation law with direct and indirect methods [33]. Thus, indirect methods estimate the plant parameters for adjusting the controller, while direct methods utilize the estimated parameters directly in the adaptive controller [2]. The indirect method is based on the certainty equivalence principle (CEP), where the model of the system is adjusted by observing its behavior through time to consequently design a control policy regarding the obtained model as veracious [34].

The model reference adaptive control (MRAC) employs a reference model (consequent with the desired behavior) to define the adaptive laws to ensure output tracking [35]. Besides, it must be considered that changes in the operating conditions may demand a restart of the adaptation procedure.

The employed architecture applies two neuro-fuzzy systems: one for the controller, and another for the plant model. Under this scheme, the plant is firstly identified, and later the training of the controller takes place. Figure 1 shows the indirect adaptive control scheme and the neuro-fuzzy systems employed. In this diagram, the section of the reference model corresponds to the system's desired behavior.

Plant identification can be performed offline to obtain data from the plant in an open loop using different input signals to represent the plant behavior; this is to set an initial neuro-fuzzy model. When the identification control system is in operation, identification is performed online to obtain the training data to measure the plant during operation.

In the adaptive control, the offline identification allows an initial configuration for the model of the plant employing the back propagation algorithm [36]. Then, as observed in Figure 1, the model is progressively adjusted online taking input and output data from the plant. Considering the scheme in Figure 1 after identification, the model is used in the control loop to train the controller using dynamic back propagation [37, 38].

The adaptive neuro-fuzzy control system process is observed in Figure 2; its implementation firstly includes plant identification, and later the neuro-fuzzy controller training. As observed in the scheme, the plant model is integrated with the control loop for training the controller.

As observed, the adaptive control process in Figure 2 includes the following steps:

Step 1: Definition of the structures and initial configurations of the controller and the neuro-fuzzy model. As an option, a controller's previous training and plant identification can be performed.

Step 2: Put into operation the control system with the actual plant and the neuro-fuzzy controller.

Step 3: Input-output data are taken during system operation.

Step 4: The collected data is used for new training (parameter adjustment) for the neuro-fuzzy system used for plant identification.

Step 5: Controller training is performed employing the adjusted model.

Step 6: If the operation of the control is unfinished, the whole process is repeated from Step 2 for the next time interval.

As observed both plant identification and controller training are carried out iteratively in a way that the plant's output reaches the desired referenced value after a variation in the plant takes place.

3. Architecture for Plant Identification

Regarding plant identification, it must be borne that an approach to the system model consists of the estimation of a neural structure capable of performing the same function [39]. For neuro-fuzzy plant identification, three samples are used for the input and two for the output. The scheme is displayed in Figure 3. The output can be seen as a non-linear function where z^{-1} represents an element of memory (delay).

From this point of view, the result is a set of output data $y_s[n]$ given by (1), where $\mathbf{H}_p$ represents the vector of parameters in the neuro-fuzzy system.

$$y_s[n] = f_p\left(u[n], u[n-1], u[n-2], y[n-1], y[n-2], \mathbf{H}_p\right). \qquad (1)$$

Regarding [26], architecture I is employed in plant identification (compact fuzzy system) using Gaussian fuzzy sets. Figure 4 illustrates an example. Such architecture is similar to that conventionally employed in radial basis neural networks [40].

Equation (2) shows the input-output expression employing the product as t-norm and using Gaussian fuzzy sets. The equation associated with the inference process is

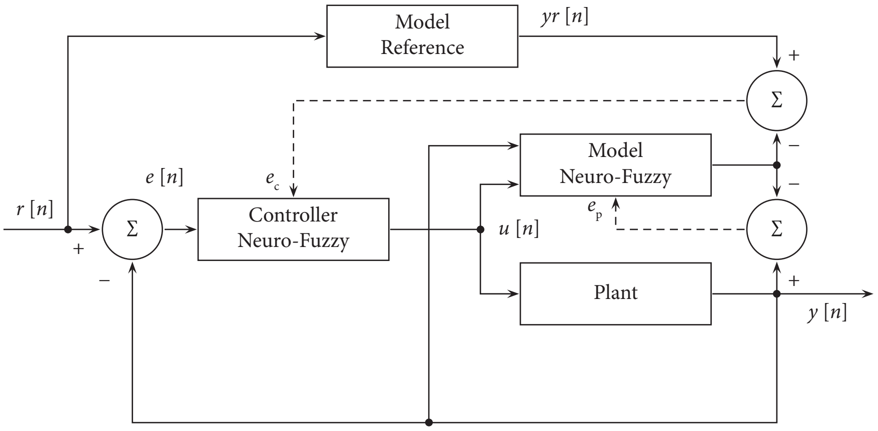


Figure 1: Control schemes using neuro-fuzzy systems.

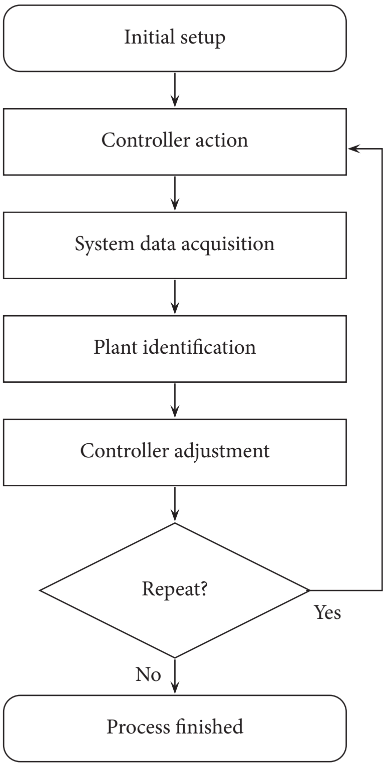


Figure 2: Algorithm for the process of adaptive control.

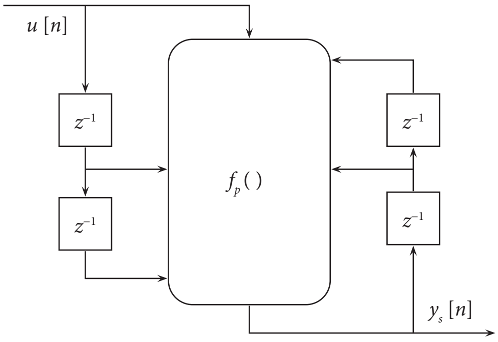


Figure 3: Neuro-fuzzy system for the plant.

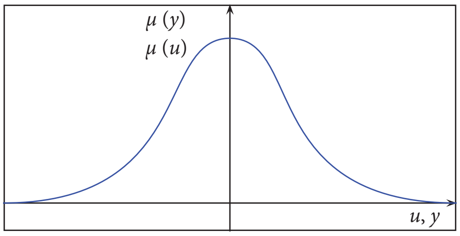


Figure 4: Gaussian fuzzy set.

$$f(w) = \sum_{l=1}^{M} y_l \left[\prod_{i=1}^{N} \exp\left(-\left(\frac{w_i - \delta_{il}}{\rho_{il}} \right)^2 \right) \right]. \tag{2}$$

In this way, the respective activation function is

$$Y_l = \prod_{i=1}^{N} \exp\left(-\left(\frac{w_i - \delta_{il}}{\rho_{il}} \right)^2 \right). \tag{3}$$

Then, the system output is calculated as

$$f = \sum_{l=1}^{M} y_l Y_l, \tag{4}$$

where M is the number of fuzzy rules, N the number of inputs to the system, y_l is the respective virtual actuator, δ_{il} and ρ_{il} are the center and the standard deviation of the Gaussian fuzzy sets. Additionally, $w_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2]\}$ is the data of input to the system; in general, all input variables are represented as w.

The representation of the associated neuro-fuzzy network can be seen in Figure 5. The first layer corresponds to the producer of the Gaussian functions, that is, the calculation of Y_l. In the second, the multiplication of Y_l and the virtual actuators y_l is performed. Finally, in the third layer, the inference output $f(w)$ is determined.

Taking into account the methodology for the design of compact fuzzy systems presented in [26] the rules that implement the structure of Figure 5 can be represented as

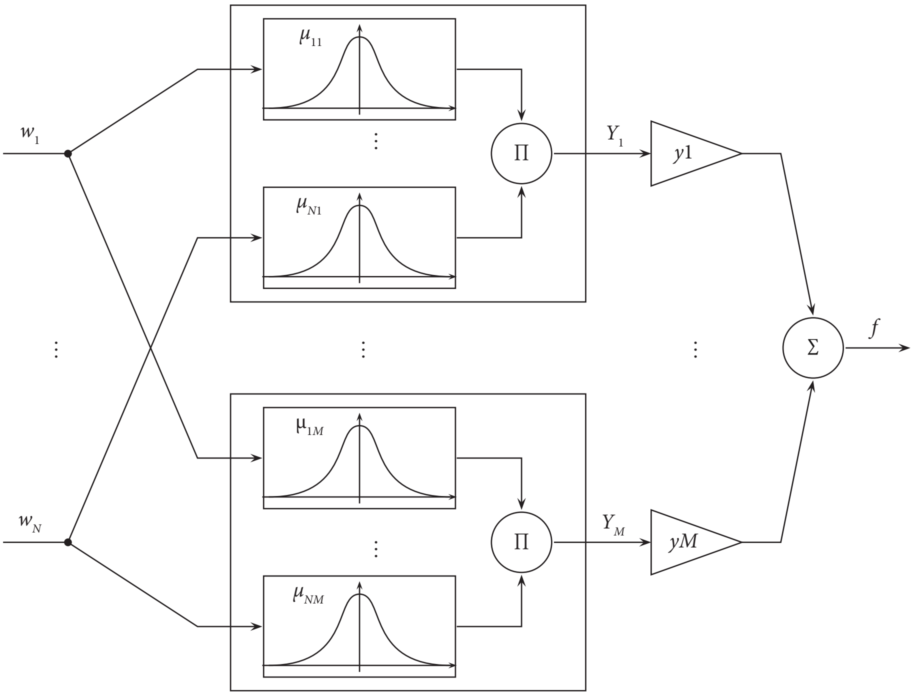


Figure 5: Representation of the neuro-fuzzy network.

shown in Boole Table 1. In general, for $l = 1, \ldots, 5$ these rules are described as follows:

If w_1 is $\mu_{1,l}$ and w_2 is $\mu_{2,l}$ and w_3 is $\mu_{3,l}$ and w_4 is $\mu_{4,l}$ and w_5 is $\mu_{5,l}$. Then the activation function is Y_l.

As observed, the activation function Y_l simultaneously depends on $\mu_{i,l}$ for $i = 1, \ldots, 5$, In this way, the output is calculated using (5) as

$$f = y_1 Y_1 + y_2 Y_2 + y_3 Y_3 + y_4 Y_4 + y_5 Y_5. \tag{5}$$

3.1. Training Equations for Plant Identification. Regarding the Back Propagation algorithm and using the descending gradient algorithm to determine the system parameters, the goal is to minimize the error corresponding to

$$J_p = \frac{1}{2}[f(w[n]) - y[n]]^2, \tag{6}$$

where $w[n]$ is the input data to the system, and $y[n]$ the desired output data for a time n. These data correspond to the respective training input-output pairs.

Taking α as the learning rate to carry out the training of the parameters, the derivatives of the error are calculated having

$$\begin{aligned} y_l(k+1) &= y_l(k) - \alpha \frac{\partial J_p}{\partial y_l}\Big|_n, \\ \delta_{il}(k+1) &= \delta_{il}(k) - \alpha \frac{\partial J_p}{\partial \delta_{il}}\Big|_n, \\ \rho_{il}(k+1) &= \rho_{il}(k) - \alpha \frac{\partial J_p}{\partial \sigma_{il}}\Big|_n. \end{aligned} \tag{7}$$

where to update the parameters, the following equations are used:

$$\begin{aligned} y_l(k+1) &= y_l(k) - \alpha(f-y)Y_l \\ \delta_{il}(k+1) &= \delta_{il}(k) - \alpha(f-y)y_l(k)Y_l \frac{2(w_i[n] - \delta_{il}(k))}{[\rho_{il}(k)]^2} \\ \rho_{il}(k+1) &= \rho_{il}(k) - \alpha(f-y)y_l(k)Y_l \frac{2(w_i[n] - \delta_{il}(k))^2}{[\rho_{il}(k)]^3}. \end{aligned} \tag{8}$$

Using the previous equations, the algorithm for training parameters of the neuro-fuzzy system can be seen in Figure 6.

Considering Figure 6, the algorithm for the adaptation (training) of parameters of the neuro-fuzzy system consists of the following steps:

Step 1: Determine the neuro-fuzzy system choosing M , N and the initial parameters.

Step 2: For the current value of $n = 1, 2, \ldots, N_T$, establish the respective input-output data pair $(w[n], y[n])$.

Step 3: Calculate the output of the neuro-fuzzy system for an input-output pair $(w[n], y[n])$, at the k -th training stage, $k = 0, 1, 2, \ldots, K_T$.

Step 4: Update parameters $y_l(k+1)$, $\delta_{il}(k+1)$, and $\rho_{il}(k+1)$, using the learning rate α.

Step 5: Return to step 2 with $k = k+1$, until the error $J_p(k)$ is smaller than a ε defined, or until k is equal to a certain number.

Table 1: Rules associated with the structure of the neuro-fuzzy system are used for identification.

$\mu_{1,1}$	$\cdots$	$\mu_{5,1}$	$\mu_{1,2}$	$\cdots$	$\mu_{5,2}$	$\mu_{1,3}$	$\cdots$	$\mu_{5,3}$	$\mu_{1,4}$	$\cdots$	$\mu_{5,4}$	$\mu_{1,5}$	$\cdots$	$\mu_{5,5}$	Y_1	Y_2	Y_3	Y_4	Y_5
1	1	1	X	X	X	X	X	X	X	X	X	X	X	X	1	0	0	0	0
X	X	X	1	1	1	X	X	X	X	X	X	X	X	X	0	1	0	0	0
X	X	X	X	X	X	1	1	1	X	X	X	X	X	X	0	0	1	0	0
X	X	X	X	X	X	X	X	X	1	1	1	X	X	X	0	0	0	1	0
X	X	X	X	X	X	X	X	X	X	X	X	1	1	1	0	0	0	0	1

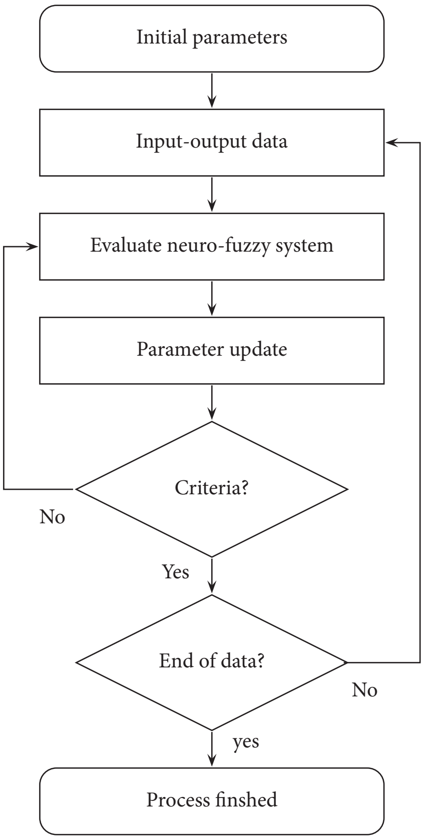


Figure 6: Algorithm for plant identification.

Step 6: Until all the data is finished, return to Step 2 with $n = n + 1$ to update the parameters using the following input-output pair $(w[n+1], y[n+1])$.

In the first step, it should be noted that the higher M is, the more parameters there are, allowing greater adaptability but with greater calculation requirements. The initial value of the parameters must also be specified, that is, $y_l(0)$, $\delta_{il}(0)$, and $\rho_{il}(0)$, which can be determined according to the linguistic rules of the expert, or chosen in a way that the membership functions uniformly cover the input values. Additionally, if the learning rate α is chosen with a very large value, it may cause the algorithm not to converge, while a very small value may cause the algorithm to require more time to reach the optimal value [7].

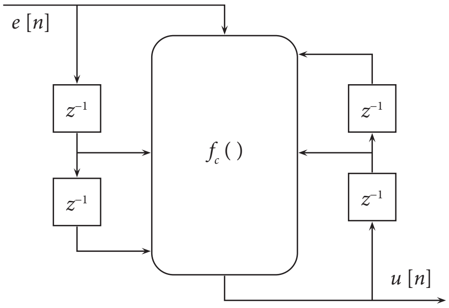


Figure 7: Input and output configuration used for the controller.

4. Controller Architecture

To implement the controller, a scheme with two input delay elements and two output delay elements is used, as can be seen in Figure 7. In this scheme, $e[n]$ is the error signal that enters the controller and $u[n]$ is the controller output corresponding to the control action.

Under this approach, the control action is given by (9), where, $\mathbf{H}_c$ corresponds to the parameter vector of the neuro-fuzzy system.

$$u[n] = f_c(u[n-1], u[n-2], e[n], e[n-1], e[n-2], \mathbf{H}_c). \tag{9}$$

The controller implementation is carried out using a compact fuzzy system with architecture II employing the fuzzy sets shown in Figure 8. Specifically, in Figure 8(a), a sigmoidal fuzzy set is employed to model positive values for the universe of discourse, while in Figure 8(b), negative values are represented for the error $e[n]$ and the control action $u[n]$,

Considering the fuzzy sets shown in Figure 8 and the general structure of the controller given by Equation 10, the scheme of Figure 9 is established, where the proposed fuzzy controller is shown.

Regarding Figure 9, the controller output can be calculated as:

$$u[n] = \sum_{i=1}^{5} \sum_{j=1}^{2} v_{ij} \mu_{ij}(x_i), \tag{10}$$

where $x_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2],\}$, For each input x_i, a function f_i can be defined as

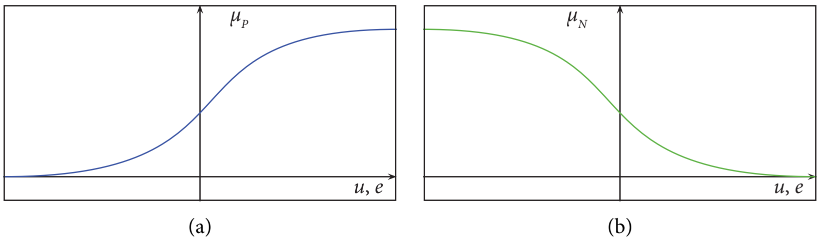


Figure 8: Fuzzy sets are employed to model positive and negative values. (a) A fuzzy set is employed to model positive values. (b) A fuzzy set is used to model negative values.

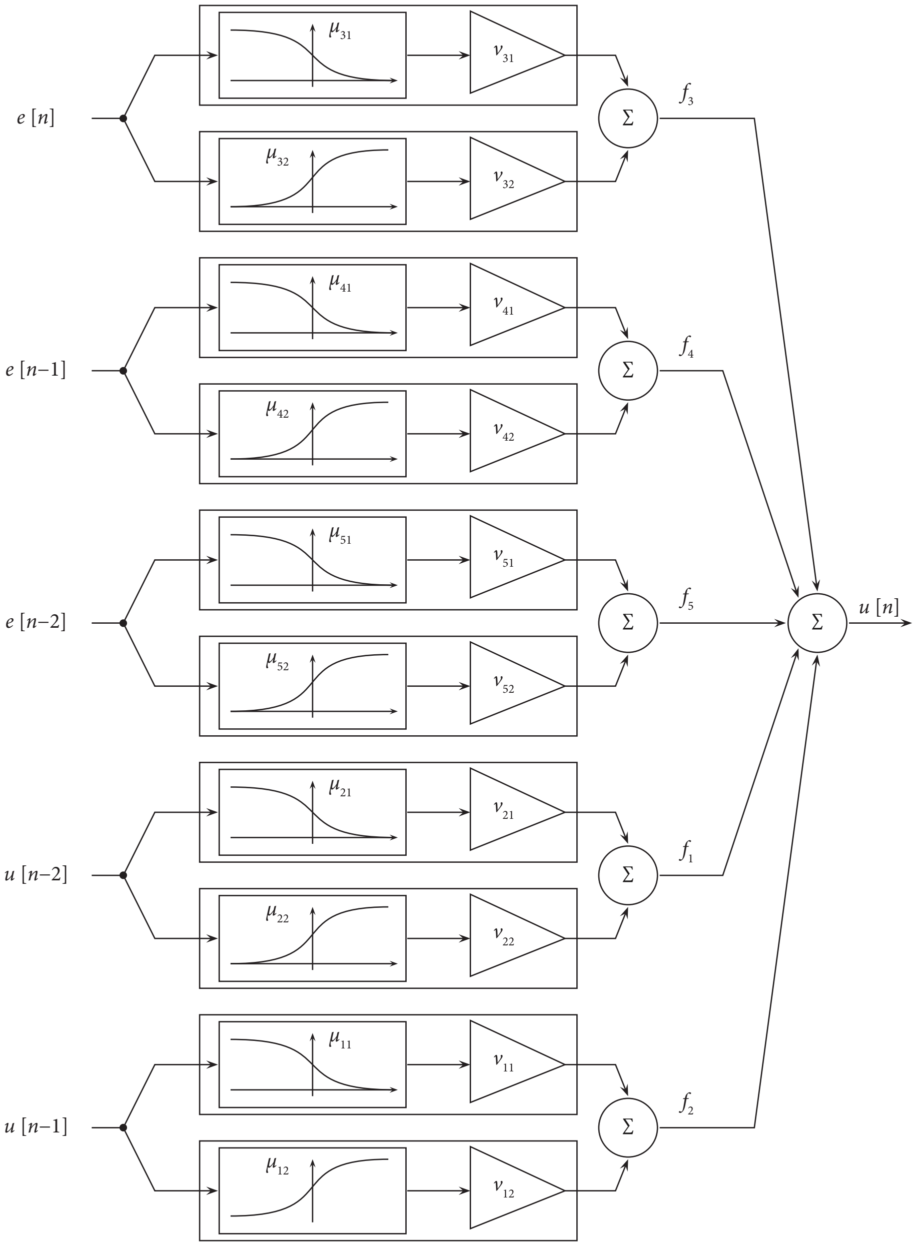


Figure 9: Scheme of the neuro-fuzzy control system.

Table 2: Rules associated with the structure of the neuro-fuzzy system used for control.

$\mu_{1,1}$	$\mu_{1,2}$	$\mu_{2,1}$	$\mu_{2,2}$	$\mu_{3,1}$	$\mu_{3,2}$	$\mu_{4,1}$	$\mu_{4,2}$	$\mu_{5,1}$	$\mu_{5,2}$	$Y_{1,1}$	$Y_{1,2}$	$Y_{2,1}$	$Y_{2,2}$	$Y_{3,1}$	$Y_{3,2}$	$Y_{4,1}$	$Y_{4,2}$	$Y_{5,1}$	$Y_{5,2}$
1	X	X	X	X	X	X	X	X	X	1	0	0	0	0	0	0	0	0	0
X	1	X	X	X	X	X	X	X	X	0	1	0	0	0	0	0	0	0	0
X	X	1	X	X	X	X	X	X	X	0	0	1	0	0	0	0	0	0	0
X	X	X	1	X	X	X	X	X	X	0	0	0	1	0	0	0	0	0	0
X	X	X	X	1	X	X	X	X	X	0	0	0	0	1	0	0	0	0	0
X	X	X	X	X	1	X	X	X	X	0	0	0	0	0	1	0	0	0	0
X	X	X	X	X	X	1	X	X	X	0	0	0	0	0	0	1	0	0	0
X	X	X	X	X	X	X	1	X	X	0	0	0	0	0	0	0	1	0	0
X	X	X	X	X	X	X	X	1	X	0	0	0	0	0	0	0	0	1	0
X	X	X	X	X	X	X	X	X	1	0	0	0	0	0	0	0	0	0	1

$$f_i = \sum_{j=1}^{2} v_{ij}\mu_{ij}(x_i), \tag{11}$$

namely,

$$f_i = v_{i1}\mu_{i1}(x_i) + v_{i2}\mu_{i2}(x_i). \tag{12}$$

Meanwhile, the membership function $\mu_{ij}(x_i)$ corresponds to

$$\mu_{ij}(x_i) = \frac{1}{1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})}}. \tag{13}$$

For derivative calculations, (13) can be represented as (14) In this way, the set of parameters of the controller corresponds to $\mathbf{H}_c = \{v_{ij}, \sigma_{ij}, \gamma_{ij}\}$ and a possible parameter of this is represented as $h_c = h_{ij} \in \mathbf{H}_c$:

$$\mu_{ij}(x_i) = \left(1 + e^{-\sigma_{ij}(x_i - \gamma_{ij})}\right)^{-1}. \tag{14}$$

Regarding the design methodology displayed in [26] for compact fuzzy systems based on Boolean relations, the rules used to implement the structure of Figure 9 can be represented as shown in Boole Table 2. In general, the rules for the fuzzy system are described as

If x_i is $\mu_{i,1}$ *Then* the activation function is $Y_{i,1}$

If x_i is $\mu_{i,2}$ *Then* the activation function is $Y_{i,2}$

where i is the index associated with the input, and j the fuzzy set negative for $j = 1$ or positive with $j = 2$, As observed, in this case, the activation function is $Y_{i,j}$ is equal to $\mu_{i,j}$ as presented in [26].

4.1. Equations for Controller Parameter Training.

The reference model (desired behavior) and the plant's neuro-fuzzy model are employed for training the neuro-fuzzy controller. Considering that y_r is the response of the reference model, firstly, the performance function used for the training process is

$$J_c = \frac{1}{2}(y_r[n] - y[n])^2 = \frac{1}{2}(e[n])^2. \tag{15}$$

Secondly, the equation associated with the plant is

$$y[n] = f_p(y[n-1], y[n-2], u[n], u[n-1], u[n-2]). \tag{16}$$

Meanwhile, the error equation is

$$e[n] = y_r[n] - y[n]. \tag{17}$$

Considering the expression for f_i , and the architecture of the controller (Figure 9), the dynamics of the controller are given by

$$u[n] = f_1(u[n-1]) + f_2(u[n-2]) + f_3(e[n]) + f_4(e[n-1]) + f_5(e[n-2]). \tag{18}$$

The derivative of the plant output with respect to the controller parameters is

$$\frac{\mathrm{d}y[n]}{\mathrm{d}h_c} = \frac{\mathrm{d}}{\mathrm{d}h_c} f_p(y[n-1], y[n-2], u[n], u[n-1], u[n-2]). \tag{19}$$

The respective derivative of the error $e[n] = y_r[n] - y[n]$ for a control parameter h_c is

$$\frac{\mathrm{d}e[n]}{\mathrm{d}h_c} = -\frac{\mathrm{d}y[n]}{\mathrm{d}h_c}. \tag{20}$$

In the same way, the derivative of the control action $u[n]$ with respect to h_{ij} is

$$\frac{\mathrm{d}u[n]}{\mathrm{d}h_{ij}} = \frac{\mathrm{d}f_1(e[n])}{\mathrm{d}h_{ij}} + \frac{\mathrm{d}f_2(e[n-1])}{\mathrm{d}h_{ij}} + \frac{\mathrm{d}f_3(e[n-2])}{\mathrm{d}h_{ij}} + \frac{\mathrm{d}f_4(u[n-1])}{\mathrm{d}h_{ij}} + \frac{\mathrm{d}f_5(u[n-2])}{\mathrm{d}h_{ij}}. \tag{21}$$

As can be seen, the respective derivatives of the parameters with respect to the plant and the controller must be calculated.

4.2. Derivatives of the Controller Parameters with respect to the Plant.

In order to have a compact expression for the plant dynamics, the representation $w_i \in \{y[n-1], y[n-2], u[n], u[n-1], u[n-2]\}$ is used. In this way, the equation to calculate the output of the plant model is

$$y = \sum_{l=1}^{M} y_l \left[\prod_{i=1}^{N} \exp\left(-\left(\frac{w_i - \delta_{il}}{\rho_{il}} \right)^2 \right) \right], \tag{22}$$

with

$$\mu_{il}(w_i) = \exp\left(-\left(\frac{w_i - \delta_{il}}{\rho_{il}} \right)^2 \right), \tag{23}$$

then

$$\begin{aligned} y &= \sum_{l=1}^{M} y_l \left[\prod_{i=1}^{N} \mu_{il}(w_i) \right], \\ y &= y_1 \prod_{i=1}^{N} \mu_{i1}(w_i) + y_2 \prod_{i=1}^{N} \mu_{i2}(w_i) + y_3 \prod_{i=1}^{N} \mu_{i3}(w_i) + \cdots. \end{aligned} \tag{24}$$

Using the auxiliary index j to consider the case where $j \neq i$, the respective derivative (for the products) is

$$\frac{\mathrm{d}}{\mathrm{d}h_c} \left(\prod_{i=1}^{N} \mu_{il}(w_i) \right) = \sum_{i=1}^{N} \left(\left(\prod_{j=1 j \neq i}^{N} \mu_{jl}(w_j) \right) \frac{\mathrm{d}\mu_{il}(w_i)}{\mathrm{d}w_i} \frac{\mathrm{d}w_i}{\mathrm{d}h_c} \right), \tag{25}$$

taking

$$P_{il} = \frac{\mathrm{d}\mu_{il}(w_i)}{\mathrm{d}w_i} \prod_{\substack{j=1 \\ \\ j \neq i}}^{N} \mu_{jl}(w_j), \tag{26}$$

with

$$\frac{\mathrm{d}\mu_{il}(w_i)}{\mathrm{d}w_i} = \frac{d}{dw_i} \left(e^{-(w_i - \delta_{il}/\rho_{il})^2} \right) = -2e^{-(w_i - \delta_{il}/\rho_{il})^2} \frac{w_i - \delta_{il}}{\rho_{il}^2}. \tag{27}$$

In this way, it is obtained:

$$\begin{aligned} \frac{\mathrm{d}y}{\mathrm{d}h_c} &= y_1 \sum_{i=1}^{N} P_{i1} \frac{\mathrm{d}w_i}{\mathrm{d}h_c} + y_2 \sum_{i=1}^{N} P_{i2} \frac{\mathrm{d}w_i}{\mathrm{d}h_c} + y_3 \sum_{i=1}^{N} P_{i3} \frac{\mathrm{d}w_i}{\mathrm{d}h_c} + \cdots, \\ \frac{\mathrm{d}y}{\mathrm{d}h_c} &= \sum_{i=1}^{N} \left(y_1 P_{i1} + y_2 P_{i2} + y_3 P_{i3} + \cdots \right) \frac{\mathrm{d}w_i}{\mathrm{d}h_c}, \\ \frac{\mathrm{d}y}{\mathrm{d}h_c} &= \sum_{i=1}^{N} \left(\sum_{l=1}^{M} y_l P_{il} \right) \frac{\mathrm{d}w_i}{\mathrm{d}h_c}, \end{aligned} \tag{28}$$

Namely:

$$\frac{\mathrm{d}y}{\mathrm{d}h_c} = \left(\sum_{l=1}^{M} y_l P_{1l} \right) \frac{\mathrm{d}w_1}{\mathrm{d}h_c} + \left(\sum_{l=1}^{M} y_l P_{2l} \right) \frac{\mathrm{d}w_2}{\mathrm{d}h_c} + \left(\sum_{l=1}^{M} y_l P_{3l} \right) \frac{\mathrm{d}w_3}{\mathrm{d}h_c} + \cdots, \tag{29}$$

with

$$C_i = \sum_{l=1}^{M} y_l P_{il}, \tag{30}$$

then, finally, it is established that

$$\frac{\mathrm{d}y[n]}{\mathrm{d}h_{ij}} = C_1 \frac{\mathrm{d}w_1}{\mathrm{d}h_{ij}} + C_2 \frac{\mathrm{d}w_2}{\mathrm{d}h_{ij}} + C_3 \frac{\mathrm{d}w_3}{\mathrm{d}h_{ij}} + C_4 \frac{\mathrm{d}w_4}{\mathrm{d}h_{ij}} + C_5 \frac{\mathrm{d}w_5}{\mathrm{d}h_{ij}}. \tag{31}$$

4.3. Derivatives of the Controller Parameters with respect to the Controller. In this procedure, the auxiliary index l is used to consider the case where $l \neq i$, thus, to determine the respective derivatives of the controller parameters considering the controller equations,

$$\frac{\mathrm{d}f_l(x_l)}{\mathrm{d}h_{ij}} = \frac{\mathrm{d}}{\mathrm{d}h_{ij}} \left(v_{l1} \mu_{l1}(x_l) \right) + \frac{\mathrm{d}}{\mathrm{d}h_{ij}} \left(v_{l2} \mu_{l2}(x_l) \right), \tag{32}$$

where $l = 1, \ldots, 5$, $i = 1, \ldots, 5$ and $j = 1, 2$; therefore, there are different cases for values of i and l, For the case when $l \neq i$

$$\begin{aligned} \frac{\mathrm{d}f_l(x_l)}{\mathrm{d}h_{ij}} &= \frac{\mathrm{d}}{\mathrm{d}x_l} \left(v_{l1} \mu_{l1}(x_l) \right) \frac{\mathrm{d}x_l}{\mathrm{d}h_{ij}} + \frac{\mathrm{d}}{\mathrm{d}x_l} \left(v_{l2} \mu_{l2}(x_l) \right) \frac{\mathrm{d}x_l}{\mathrm{d}h_{ij}} \\ \frac{\mathrm{d}f_l(x_l)}{\mathrm{d}h_{ij}} &= \left[\frac{\mathrm{d}}{\mathrm{d}x_l} \left(v_{l1} \mu_{l1}(x_l) \right) + \frac{\mathrm{d}}{\mathrm{d}x_l} \left(v_{l2} \mu_{l2}(x_l) \right) \right] \frac{\mathrm{d}x_l}{\mathrm{d}h_{ij}}. \end{aligned} \tag{33}$$

For $j = 1.2$, the respective derivatives are

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}x_l} \left(v_{lj} \mu_{lj}(x_l) \right) &= \frac{\mathrm{d}}{\mathrm{d}x_l} \left(v_{lj} \left(1 + e^{-\sigma_{lj}(x_l - \gamma_{lj})} \right)^{-1} \right), \\ \frac{\mathrm{d}}{\mathrm{d}x_l} \left(v_{lj} \mu_{lj}(x_l) \right) &= v_{lj} \left(1 + e^{-\sigma_{lj}(x_l - \gamma_{lj})} \right)^{-2} e^{-\sigma_{lj}(x_l - \gamma_{lj})} \sigma_{lj}. \end{aligned} \tag{34}$$

On the other hand, when $l = i$ and $j = 1$

$$\frac{df_i(x_i)}{dh_{i1}} = \frac{d}{dh_{i1}}\left(\nu_{i1}\mu_{i1}(x_i)\right) + \frac{d}{dx_i}\left(\nu_{i2}\mu_{i2}(x_i)\right)\frac{dx_i}{dh_{i1}}. \tag{35}$$

Also, when $l = i$ and $j = 2$, the respective derivatives are

$$\frac{df_i(x_i)}{dh_{i2}} = \frac{d}{dx_i}\left(\nu_{i1}\mu_{i1}(x_i)\right)\frac{dx_i}{dh_{i2}} + \frac{d}{dh_{i2}}\left(\nu_{i2}\mu_{i2}(x_i)\right). \tag{36}$$

In order to develop the above equations, first:

$$\frac{d}{dx_i}\left(\nu_{ij}\mu_{ij}(x_i)\right) = \nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\sigma_{ij}. \tag{37}$$

Second, for the other derivatives,

$$\frac{d}{dh_{ij}}\left(\nu_{ij}\mu_{ij}(x_i)\right) = \frac{d}{dh_{ij}}\left(\nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-1}\right). \tag{38}$$

For the parameter $h_{ij} = \nu_{ij}$,

$$\frac{d}{d\nu_{ij}}\left(\nu_{ij}\mu_{ij}(x_i)\right) = \left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-1} - \nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\left(-\sigma_{ij}\right)\frac{dx_i}{d\nu_{ij}}. \tag{39}$$

Taking the parameter $h_{ij} = \sigma_{ij}$, it is established that

$$\frac{d}{d\sigma_{ij}}\left(\nu_{ij}\mu_{ij}(x_i)\right) = -\nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\left(\gamma_{ij} - x_i - \sigma_{ij}\frac{dx_i}{d\sigma_{ij}}\right). \tag{40}$$

With the parameter $h_{ij} = \gamma_{ij}$, it is determined that:

$$\frac{d}{d\gamma_{ij}}\left(\nu_{ij}\mu_{ij}(x_i)\right) = -\nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\left(\sigma_{ij} - \sigma_{ij}\frac{dx_i}{d\gamma_{ij}}\right). \tag{41}$$

In general, these equations can be written as

$$\begin{aligned}
\frac{d}{d\nu_{ij}}\left(\nu_{ij}\mu_{ij}(x_i)\right) &= F_{\nu_{ij}} + K_{\nu_{ij}}\frac{dx_i}{d\nu_{ij}},\\
\frac{d}{d\sigma_{ij}}\left(\nu_{ij}\mu_{ij}(x_i)\right) &= F_{\sigma_{ij}} + K_{\sigma_{ij}}\frac{dx_i}{d\sigma_{ij}},\\
\frac{d}{d\gamma_{ij}}\left(\nu_{ij}\mu_{ij}(x_i)\right) &= F_{\gamma_{ij}} + K_{\gamma_{ij}}\frac{dx_i}{d\gamma_{ij}},
\end{aligned} \tag{42}$$

where

$$\begin{aligned}
F_{\nu_{ij}} &= \left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-1},\\
F_{\sigma_{ij}} &= -\nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\left(\gamma_{ij} - x_i\right),\\
F_{\gamma_{ij}} &= -\nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\sigma_{ij},\\
K_{\nu_{ij}} &= \nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\sigma_{ij},\\
K_{\sigma_{ij}} &= \nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\sigma_{ij},\\
K_{\gamma_{ij}} &= \nu_{ij}\left(1 + e^{-\sigma_{ij}(x_i-\gamma_{ij})}\right)^{-2} e^{-\sigma_{ij}(x_i-\gamma_{ij})}\sigma_{ij}.
\end{aligned} \tag{43}$$

In general, considering a parameter h_{ij} , the derivatives can be represented in the form

$$\frac{d}{dh_{ij}}\left(v_{ij}\mu_{ij}\left(x_i\right)\right) = F_{h_{ij}} + K_{h_{ij}}\frac{dx_i}{dh_{ij}}. \tag{44}$$

Regarding the case when $l = i$ and $j = 1$ of the, it is obtained:

$$\begin{aligned}\frac{df_i(x_i)}{dh_{i1}} &= \frac{d}{dh_{i1}}\left(v_{i2}\mu_{i2}(x_i)\right) + \frac{d}{dx_i}\left(v_{i2}\mu_{i2}(x_i)\right)\frac{dx_i}{dh_{i1}} \\ &= F_{h_{i1}} + K_{h_{i1}}\frac{dx_i}{dh_{i1}} + \frac{d}{dx_i}\left(v_{i2}\mu_{i2}(x_i)\right)\frac{dx_i}{dh_{i1}} \\ &= F_{h_{i1}} + \left(K_{h_{i1}} + \frac{d}{dx_i}\left(v_{i2}\mu_{i2}(x_i)\right)\right)\frac{dx_i}{dh_{i1}}.\end{aligned} \tag{45}$$

Also, when $l = i$ and $j = 2$ in it is established that

$$\begin{aligned}\frac{df_i(x_i)}{dh_{i2}} &= \frac{d}{dx_i}\left(v_{i1}\mu_{i1}(x_i)\right)\frac{dx_i}{dh_{i2}} + \frac{d}{dh_{i2}}\left(v_{i2}\mu_{i2}(x_i)\right) \\ &= \frac{d}{dx_i}\left(v_{i2}\mu_{i2}(x_i)\right)\frac{dx_i}{dh_{i2}} + F_{h_{i2}} + K_{h_{i2}}\frac{dx_i}{dh_{i2}} \\ &= F_{h_{i2}} + \left(\frac{d}{dx_i}\left(v_{i2}\mu_{i2}(x_i)\right) + K_{h_{i2}}\right)\frac{dx_i}{dh_{i2}}.\end{aligned} \tag{46}$$

4.4. Controller Training Process. Taking $w_i \in \{y[n-1], y[n-2], u[n], u[n-1], u[n-2]\}$ and $x_i \in \{u[n-1], u[n-2], e[n], e[n-1], e[n-2]\}$, in general the equations that implement the dynamics of the training parameters are

$$\begin{aligned}&\frac{de[n]}{dh_{ij}} = -\frac{dy}{dh_{ij}}, \\ &\frac{dy[n]}{dh_{ij}} = C_1\frac{dw_1}{dh_{ij}} + C_2\frac{dw_2}{dh_{ij}} + C_3\frac{dw_3}{dh_{ij}} + C_4\frac{dw_4}{dh_{ij}} + C_5\frac{dw_5}{dh_{ij}}, \\ &\frac{du[n]}{dh_{ij}} = \frac{df_1(x_1)}{dh_{ij}} + \frac{df_2(x_2)}{dh_{ij}} + \frac{df_3(x_3)}{dh_{ij}} + \frac{df_4(x_4)}{dh_{ij}} + \frac{df_5(x_5)}{dh_{ij}}.\end{aligned} \tag{47}$$

In the last equation, if $l \neq i$, then,

$$\frac{df_l(x_l)}{dh_{ij}} = \left[\frac{d}{dx_l}\left(v_{l1}\mu_{l1}(x_l)\right) + \frac{d}{dx_l}\left(v_{l2}\mu_{l2}(x_l)\right)\right]\frac{dx_l}{dh_{ij}}. \tag{48}$$

On the other hand, if $l = i$, it is determined that

$$\begin{aligned}\frac{df_i(x_i)}{dh_{i1}} &= F_{h_{i1}} + \left(K_{h_{i1}} + \frac{d}{dx_i}\left(v_{i2}\mu_{i2}(x_i)\right)\right)\frac{dx_i}{dh_{i1}}, \\ \frac{df_i(x_i)}{dh_{i2}} &= F_{h_{i2}} + \left(\frac{d}{dx_i}\left(v_{i1}\mu_{i1}(x_i)\right) + K_{h_{i2}}\right)\frac{dx_i}{dh_{i2}}.\end{aligned} \tag{49}$$

Finally, the following equation is employed to update the parameters:

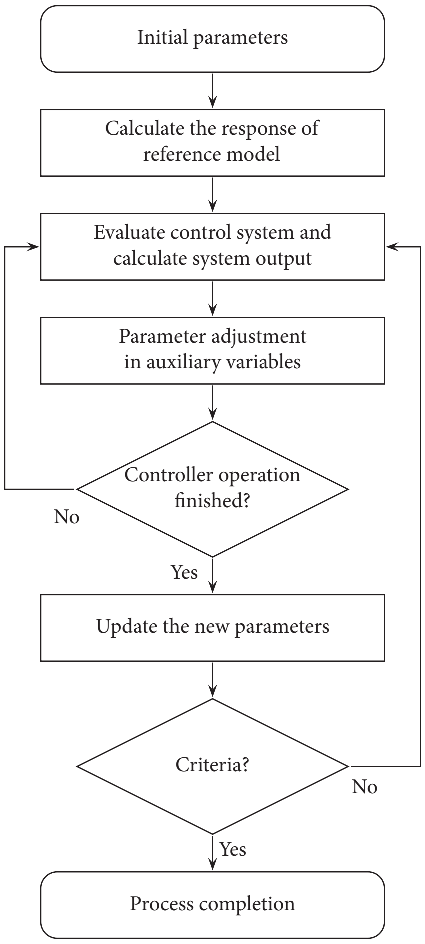

Initial parameters
Calculate the response of reference model
Evaluate control system and calculate system output
Parameter adjustment in auxiliary variables
Controller operation finished?
No
Yes
Update the new parameters
Criteria?
No
Yes
Process completion

Figure 10: Algorithm for training the neuro-fuzzy controller.

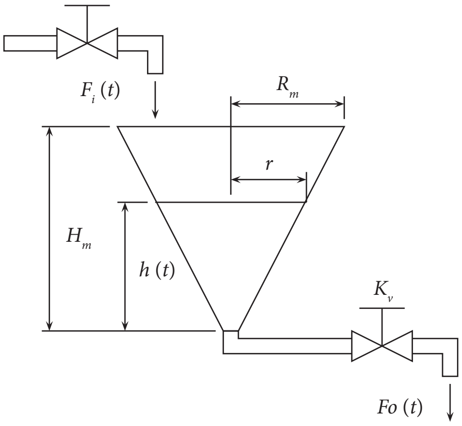


Figure 11: Conical tank diagram.

$$h_{ij}(k+1) = h_{ij}(k) - \alpha e[n]\frac{de[n]}{dh_{ij}}, \tag{50}$$

where α corresponds to the learning rate. Considering each parameter, it is obtained that

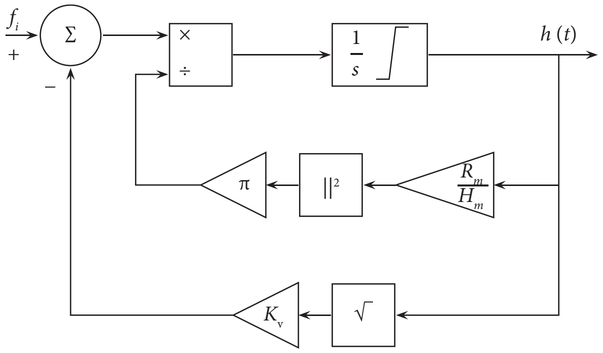


FIGURE 12: Block diagram associated with the conical tank.

TABLE 3: Conical tank parameters.

Parameter	Description	Value
R_m	Max radius	$0.48m$
H_m	Max height	$1.2m$
F_i	Input flow	0.0072 m^3/s (max)
K_v	Valve coefficient	$0.0067m^2/s$
h	Height	Variable
F_o	Outflow	Variable
r	Radius dependent of h	Variable

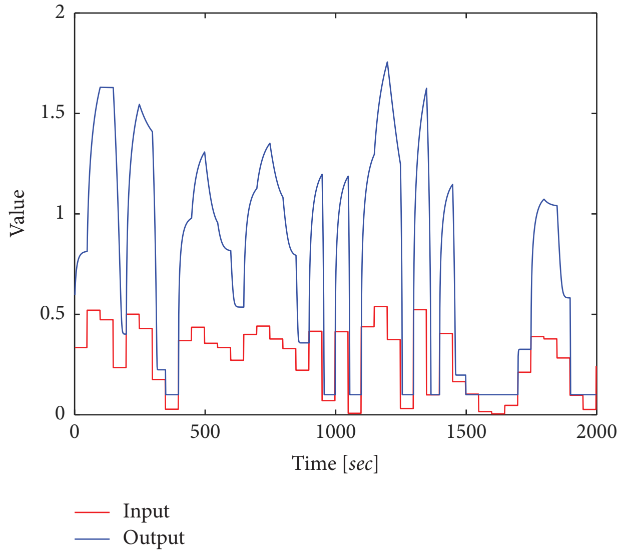


FIGURE 13: Data of the plant in an open loop.

$$v_{ij}(k+1) = v_{ij}(k) - \alpha e[n]\frac{de[n]}{dv_{ij}},$$

$$\sigma_{ij}(k+1) = \sigma_{ij}(k) - \alpha e[n]\frac{de[n]}{d\sigma_{ij}}, \qquad (51)$$

$$\gamma_{ij}(k+1) = \gamma_{ij}(k) - \alpha e[n]\frac{de[n]}{d\gamma_{ij}}.$$

Considering the previous equations in Figure 10, the algorithm for training the neuro-fuzzy controller can be observed.

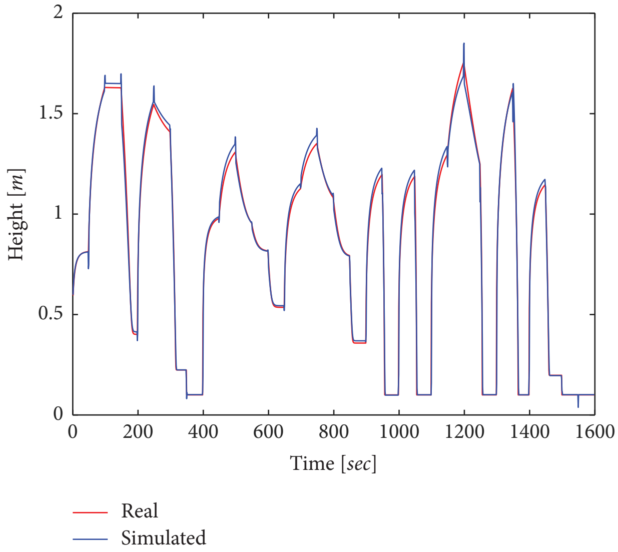


FIGURE 14: Result of the identification process in open loop.

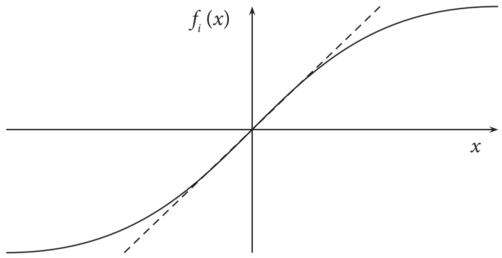


FIGURE 15: Considered shape for f_i.

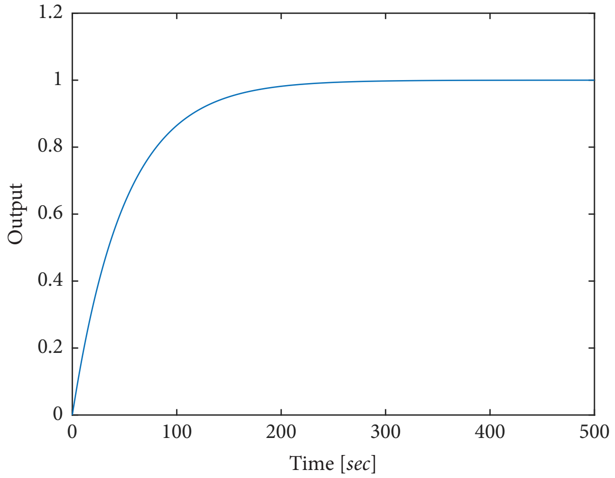
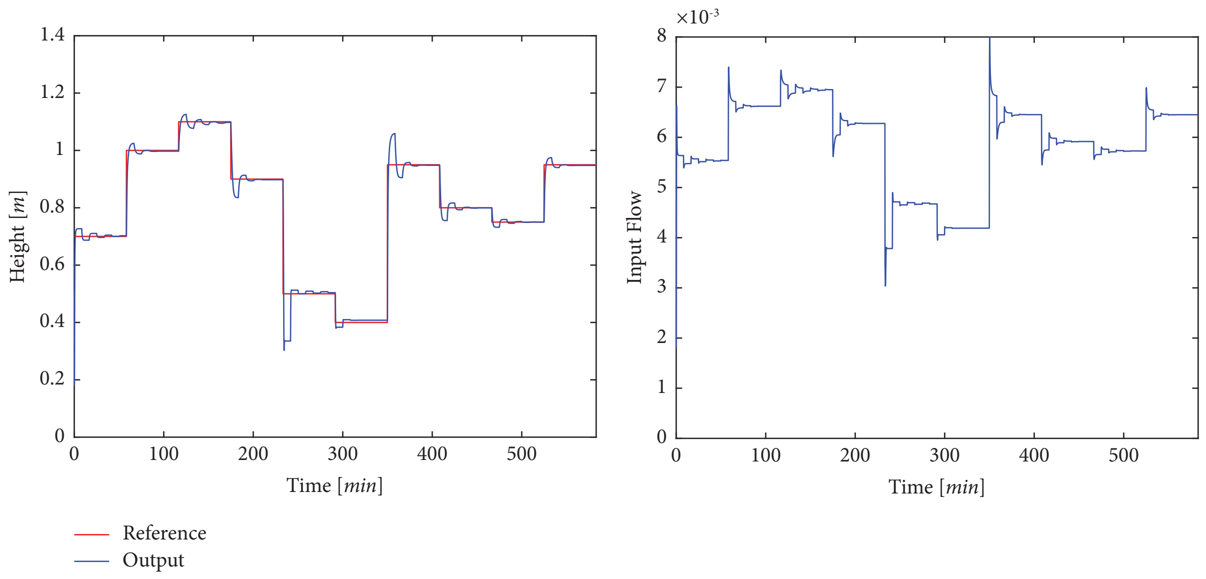


FIGURE 16: Reference model response.

Regarding the flowchart in Figure 10, the steps of the algorithm used to train the neuro-fuzzy controller are as follows:

Step 1: Establish the plant model and choose the initial configuration of the controller parameters.



(a) (b)

Figure 17: Response of the adaptive control system. (a) Control system output. (b) Input flow to the conical tank.

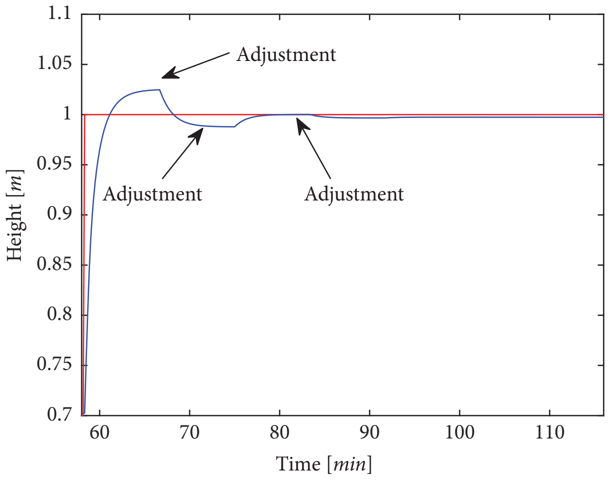


Figure 18: Detail of the adaptive control system response.

Step 2: Calculate the reference model response y_r (desired response).

Step 3: In the k -th training stage, $k = 0, 1, 2, \ldots, K_T$, for the current n value of simulation, the output of the control system is calculated using the neuro-fuzzy model of the plant.

Step 4: Carry out the parameter adjustment of the neuro-fuzzy controller (in auxiliary variables) using the respective equations that involve the dynamics of the control system, and the respective derivatives (equations (51), (54), and subsequent).

Step 5: Until n is equal to the defined value N_T (simulation time), return to Step 3 for the next simulation step $n = n + 1$ (where the output of the control system is calculated).

Step 6: In the case of completing the simulation time, the newly optimized controller parameters are updated.

Step 7: Return to Step 3 for a new iteration $k = k + 1$, until $J_c(k) = \sum_{n=1}^{N_T} J_c(k, n)$ is smaller than a ε defined, or until k is equal to a defined number K_T.

It should be noted that in Step 4 of the algorithm the parameters to be adjusted are stored in auxiliary variables, since during this step the controller parameters do not change.

5. Conical Tank Model

The conical tank system is a single input single output (SISO) process. The output of this process is the level $h(t)$ and the input to the process is the liquid flow F_i , The scheme of the conical tank considered can be seen in Figure 11.

For this system, a liquid with constant density enters at a volumetric rate F_i in a conical tank of height H_m and maximum radius R_m , The output of the tank is $F_0 = K_v\sqrt{h}$, where h is the height of the liquid in the tank, and K_v is the valve coefficient. The process has high non-linearity due to changes in the process gain, and the time constant with respect to the height of the liquid in the tank [41].

Considering the input F_i, and the output F_o, according to the law of mass conservation, the accumulation is the mass that enters minus the mass that leaves. Since mass conservation manifests as an increase or decrease in volume, accumulation is the change in volume with respect to time.

$$\frac{dV}{dt} = F_i - F_o. \tag{52}$$

Considering Figure 11, the circular cone has volume:

$$V = \frac{\pi}{3} r^2 h. \tag{53}$$

From the geometry of the tank, it is observed that:

$$r = \frac{R_m}{H_m} h. \tag{54}$$

Then:

$$V = \frac{\pi}{3}\left(\frac{R_m}{H_m}\right)^2 h^3. \tag{55}$$

Taking the derivative with respect to time:

$$\frac{dV}{dt} = \pi\left(\frac{R_m}{H_m}\right)^2 h^2 \frac{dh}{dt}. \tag{56}$$

In this way,

$$\frac{dh}{dt} = \frac{F_i - K_v h}{\pi (R_m/H_m)^2 h^2}. \tag{57}$$

The block diagram of this system is shown in Figure 12, where the non-linearities present in the hydraulic system can be seen. Tank parameters are displayed in Table 3.

6. Results

A systematic location of the Gaussian membership functions in the range [−1.5 1.5] with variance 2, and virtual actuators started in zero, is carried out for the fuzzy system initial configuration (employed for plant identification). As this assignment is systematic, it requires previous training before using it in the adaptive control system. The data used to perform the training can be seen in Figure 13.

Plant identification is performed by using 400 training epochs, and the results can be seen in Figure 14 obtaining a mean squared error (MSE) of 0.0012.

For the initial configuration of the controller are employed functions f_i displaying a linear behavior in the range [−2.0 2.0], as shown in Figure 15; in addition, virtual actuators are started in zero for parameter optimization.

The reference model given by the transfer function in is used to carry out controller training. It can be observed that the system has a settling time of 200 1seconds. Figure 16 shows the reference model response, namely, the type of expected (desired) behavior when optimizing the controller.

$$G(s) = \frac{Y(s)}{R(s)} = \frac{1}{50s + 1}. \tag{58}$$

The simulation of the adaptive control is achieved considering different reference values where the identification of the plant and the training of the controller are carried out iteratively. The simulation result is shown in Figure 17, where Figure 17(a) displays the level of the tank when the reference value is changed. In the first iteration, the plant output does not reach the reference value, after several iterations for the plant identification and controller optimization, the plant output reaches the reference value. In this way, it can be seen that the adaptive process allows adjusting the neuro-fuzzy model of the plant and the controller. Meanwhile, Figure 17(b) displays the input flow to the conical tank in, which is required for the system to reach the reference value. In this result, a progressive adjustment is also observed for the input flow to achieve the desired output value.

Furthermore, to observe the operation of the adaptive control system in detail, Figure 18 shows progressive adjustments to identify the plant and the controller training; in this way, it is observed that the tank level reaches the reference value. When each adjustment is made, the dynamic system response displays similar behavior to the reference model. Plant identification and controller adjustment are carried out with data obtained during the operation of the plant. In this way, it is observed the correction made by the system when the controller (optimized) comes into operation. Finally, it can be also seen that the greatest adjustment is made in the first iteration of the iterative process.

7. Discussion

Regarding [26] in this work, architecture I was used for plant identification and architecture II for the controller. As can be seen, neuro-fuzzy architecture I present greater complexity than architecture II, whereby previous training for the neuro-fuzzy system was necessary. It is possible to define a strategy to reduce preliminary information aiming at reducing the complexity when settling inference rules in the neuro-fuzzy system.

In addition, the distribution of Gaussian sets in architecture I is relevant for the initial configuration. In this development, the distribution was carried out systematically in the respective input universes; nevertheless, different types of fuzzy sets and other alternatives can be considered as part of the assignment and configuration.

The development presented in this work may be employed to implement MIMO systems, such as that presented in [28], since MIMO systems can be built from SISO sub-systems with neuro-fuzzy architectures of types I and II.

Plant identification is essential when implementing adaptive control, which is rooted in the equivalence principle of certainty. To improve the performance of the control system, it is first necessary to guarantee a suitable identification process. Further works may well consider different configurations and architectures in the membership functions to design neuro-fuzzy systems.

The work presented here consists of an application of fuzzy systems based on Boolean relations considering the most relevant aspects in regard to the design and application of an adaptive neuro-fuzzy control system. Regarding developments of the proposed adaptive control system, the implementations developed have mainly focused on displaying different aspects related to the design and training considering various plant types. Consequently, it is expected to carry out a detailed stability analysis of fuzzy systems based on Boolean relations using the Lyapunov functions and the Krasovskii theorem in later work.

8. Conclusions

The proposed adaptive neuro-fuzzy control scheme allows the iterative adjustment of the plant model and the controller when there are variations in the plant. Additionally, this work displays the necessary equations and algorithms for training neuro-fuzzy systems.

Plant identification is essential for the correct operation of the adaptive control system; for this reason, the initial configuration of the plant model takes place considering the equivalence principle of certainty. Likewise, for plant identification, the amount of available data is limited when having a variety of parameters, which requires progressive adjustment to identify the plant and to optimize the controller.

The simulations allow us to observe that the adaptive scheme achieves the adjustment of the plant model and controller with the aim that the system output reaches the expected (desired) value.

Further works may consider different configurations and architectures of fuzzy sets, including the mechanisms to determine the initial configuration of neuro-fuzzy systems, aiming to improve the adaptive control system performance.

Moreover, SISO sub-systems could be employed with neuro-fuzzy architectures I and II to enhance the implementation of adaptive control MIMO systems.

Data Availability

The simulation data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Universidad Distrital Francisco José de Caldas and in part by the Universidad de Oviedo. The authors express gratitude to the Universidad Distrital Francisco José de Caldas and Universidad de Oviedo.

References

[1] I. D. Landau, R. Lozano, M. M'Saad, and A. Karimi, *Adaptive Control, Algorithms, Analysis and Applications*, Springer, Berlin, Germany, 2011.

[2] C. Cao, L. Ma, and Y. Xu, "Adaptive control theory and applications," *Journal of Control Science and Engineering*, vol. 2012, p. 201, Article ID 827353, 2012.

[3] K. Chen and A. Astolfi, "Adaptive control for nonlinear systems with time-varying parameters and control coefficient," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 3829–3834, 2020.

[4] K. Wang, X. Li, and Y. Li, "Multiple model adaptive tracking control based on adaptive dynamic programming," *Discrete Dynamics in Nature and Society*, vol. 2016, pp. 1–12, Article ID 6023892, 2016.

[5] Y. Li, Y. Yin, and S. Zhang, "Adaptive control of delayed teleoperation systems with parameter convergence," *Mathematical Problems in Engineering*, vol. 18, p. 45, Article ID 1046419, 2018.

[6] K. Passino, *Biomimicry for Optimization, Control, and Automation*, Springer-Verlag, London, UK, 2005.

[7] H. T. Nguyen, N. R. Prasad, C. L. Walker, and E. A. Walker, "A first course in FUZZY and NEURAL CONTROL," *CHAPMAN & HALL/CRC*, vol. 34, p. 286, 2003.

[8] B. Shi, J. Yuan, and C. Dong, "On fractional model reference adaptive control," *The Scientific World Journal*, vol. 2014, pp. 1–8, Article ID 521625, 2014.

[9] A. Rincon and F. Angulo, "Adaptive control for nonlinear systems with time-varying control gain," *Journal of Control Science and Engineering*, vol. 2012, pp. 1–9, Article ID 269346, 2012.

[10] Z. Zhang, Z. Yang, S. Xiong, S. Chen, S. Liu, and X. Zhang, "Simple adaptive control-based reconfiguration design of Cabin pressure control system," *Complexity*, vol. 2021, pp. 1–16, Article ID 6635571, 2021.

[11] X. Zuo, J. W. Liu, X. Wang, and H. Q. Liang, "Adaptive PID and model reference adaptive control switch controller for nonlinear hydraulic actuator," *Mathematical Problems in Engineering*, vol. 2017, pp. 1–15, Article ID 6970146, 2017.

[12] J. C. Tudón-Martínez and R. Morales-Menendez, "Adaptive vibration control system for MR damper faults," *Shock and Vibration*, vol. 2015, pp. 1–17, Article ID 163694, 2015.

[13] I. Machón-González and H. López-García, "Feedforward nonlinear control using neural gas network," *Complexity*, vol. 2017, pp. 1–11, Article ID 3125073, 2017.

[14] W. Guan, H. Zhou, Z. Su, X. Zhang, and C. Zhao, "Ship steering control based on Quantum neural network," *Complexity*, vol. 2019, pp. 1–10, Article ID 3821048, 2019.

[15] H. Gao, X. Li, C. Gao, and J. Wu, "Neural network supervision control strategy for inverted pendulum tracking control," *Discrete Dynamics in Nature and Society*, vol. 2021, pp. 1–14, Article ID 5536573, 2021.

[16] S. Slama, A. Errachdi, and M. Benrejeb, "Neural adaptive PID and neural indirect adaptive control switch controller for nonlinear MIMO systems," *Mathematical Problems in Engineering*, vol. 2019, pp. 1–11, Article ID 7340392, 2019.

[17] Z. Xiu-yu, L. Cui-ping, W. Jian-guo, and L. Yan, "Adaptive dynamic surface control for generator excitation control system," *Mathematical Problems in Engineering*, vol. 2014, pp. 1–11, Article ID 481936, 2014.

[18] Y. Sun, J. Xu, G. Lin, and N. Sun, "Adaptive neural network control for maglev vehicle systems with time-varying mass and external disturbance," *Neural Computing & Applications*, vol. 47, p. 486, 2021.

[19] J. Zhou and Q. Zhang, "Adaptive fuzzy control of uncertain robotic manipulator," *Mathematical Problems in Engineering*, vol. 2018, pp. 1–10, Article ID 4703492, 2018.

[20] Q. Du, L. Sha, W. Shi, and L. Sun, "Adaptive fuzzy path tracking control for mobile robots with unknown control direction," *Discrete Dynamics in Nature and Society*, vol. 2021, pp. 1–7, Article ID 9935271, 2021.

[21] H. Zhang, "Fuzzy adaptive control of uncertain MIMO chaotic systems with unknown control direction," *Complexity*, vol. 2021, pp. 1–9, Article ID 9914288, 2021.

[22] Y. Hao, S. Li, Q. Xia, and M. Wang, "Type-2 combined T-S adaptive fuzzy control," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–15, Article ID 3479389, 2020.

[23] P. Pei, Z. Pei, Z. Tang, and H. Gu, "Position tracking control of PMSM based on fuzzy PID-variable structure adaptive

control," *Mathematical Problems in Engineering*, vol. 2018, pp. 1–15, Article ID 5794067, 2018.

[24] Y. Sun, J. Xu, H. Qiang, and G. Lin, "Adaptive neural-fuzzy robust position control scheme for maglev train systems with experimental verification," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 11, pp. 8589–8599, 2019.

[25] H. Espitia, J. Soriano, I. Machón, and H. López, "Design methodology for the implementation of fuzzy inference systems based on boolean relations," *Electronics*, vol. 8, no. 11, p. 1243, 2019.

[26] H. Espitia, J. Soriano, I. Machón, and H. López, "Compact fuzzy systems based on boolean relations," *Applied Sciences*, vol. 11, no. 4, p. 1793, 2021.

[27] H. E. Espitia, I. Machón-González, H. López-García, and G. Díaz, "Proposal of an adaptive neurofuzzy system to control flow power in distributed generation systems," *Complexity*, vol. 176, p. 583, 2019.

[28] H. Espitia, I. Machón, and H. López, "Control of a MIMO coupled plant using a neuro-fuzzy adaptive system based on boolean relations," *IEEE Access*, vol. 9, pp. 59987–60009, 2021.

[29] H. Espitia, I. Machón, and H. López, "Optimization of a fuzzy automatic voltage controller using real-time recurrent learning," *Processes*, vol. 9, no. 6, pp. 947–2021, 2021.

[30] H. Espitia, I. Machón, and H. López, "Design and optimization of a neuro-fuzzy system for the control of an electromechanical plant," *Applied Sciences*, vol. 12, no. 2, p. 541, 2022.

[31] A. Astolfi, D. Karagiannis, and R. Ortega, "Nonlinear and adaptive control with applications, communications and control engineering," *Springer-Verlag London*, vol. 184, p. 44, 2007.

[32] F. Rodrígez and M. López, *Control Adaptativo Y Robusto*, Universidad de Sevilla, Sevilla, España, 1996.

[33] R. Patel, D. Deb, R. Dey, and E. Valentina, "Balas, introduction to adaptive control, adaptive and intelligent control of microbial fuel cells," *Intelligent Systems Reference Library*, Springer, vol. 161, p. 14, Cham, 2020.

[34] J. Bernat, J. Kolota, and S. Stepien, "The proportional derivative indirect adaptive control," *IFAC Proceedings Volumes*, vol. 46, no. 11, pp. 523–528, 2013.

[35] G. Tao and G. Song, "Higher order tracking properties of model reference adaptive control systems," *IEEE Transactions on Automatic Control*, vol. 63, no. 11, pp. 3912–3918, 2018.

[36] M. Singh, I. Singh, and A. Verma, "Identification on non linear series-parallel model using neural network," *MIT International Journal of Electrical and Instrumentation Engineering*, vol. 3, no. 1, pp. 21–23, 2013.

[37] O. De Jesús and M. T. Hagan, "Backpropagation algorithms for a broad class of dynamic networks," *IEEE Transactions on Neural Networks*, vol. 18, no. 1, pp. 14–27, 2007.

[38] M. N. S. S. Ke-Lin Du, *Neural Networks and Statistical Learning*, Springer, London, 2013.

[39] A. Alshejari, *Neuro-Fuzzy Based Intelligent Approaches to Nonlinear System Identification and Forecasting*, PhD Thesis, University of Westminster, London, United Kingdom, 2018.

[40] W. U. Yue, H. Wang, B. Zhang, and K.-L. Du, "Using Radial Basis Function Networks for Function Approximation and Classification," *International Scholarly Research Notices*, vol. 2012, Article ID 324194, 34 pages, 2012.

[41] S. K. Vavilala, V. Thirumavalavan, and K. Chandrasekaran, "Level control of a conical tank using the fractional order controller," *Computers & Electrical Engineering*, vol. 87, Article ID 106690, 2020.