Online Tuning of the Fuzzy PID Controller using the Back-Propagation Algorithm

Eduardo Reyes[1], Juan M.J. Ramos[2], José I. Hernandez[3], Jonam L. Sanchez[4], Gerardo M. Méndez[5]*

Instituto Tecnológico de Nuevo León, Av. Eloy Cavazos # 2001, Col. Tolteca, Cd. Guadalupe, N.L. México, CP 67170

Abstract— *This paper presents a novel methodology for the online update the fuzzy rule base of the type-1 fuzzy logic system that estimates the proportional-integral-derivative (PID) gains of the professional PID control. Two different types of benchmarking PID controllers are used to compare the performance of the proposed methodology. The first controller is the so-called professional PID (P-PID), where the proportional gain KP, the integral gain KI, and the derivative gain KD, are offline calculated based on the dynamics of the process under control using the Zeigler Nichols method: in this controller the three gains remains fixed during the entire process control. The second controller uses three type-1 fuzzy logic systems to estimate each one of the gains of the professional PID controller every control cycle; each fuzzy rule base is offline estimated by the expert and remains fixed during the complete control process. This paper proposes a fuzzy self-tuning professional PID controller: it has three singleton type-1 fuzzy logic systems to calculate each gain of the controller every control cycle, with the novel characteristics that each fuzzy rule base is updated and tuned each feedback cycle using the back-propagation BP algorithm. This proposal is named the fuzzy professional proportional-integral-derivative controller (T1 SFLS P-PID) with back-propagation (BP) tuning. The experiments show that the proposed fuzzy self-tuning controller has better transient performance compared with the two benchmarking controllers. It shows the minimum overshoot and the minimum response time.*

Keywords—- PID control, singleton type-1 fuzzy logic systems, back propagation algorithm.

I. INTRODUCTION

In [1] a fuzzy PID controller is used to reduce the overstress that arise in the actuators so as improve in a considerable way the speed of response. The work done in [2] considers different variables that influence directly or indirectly in the process by which different types of controllers are implemented to counter the effects of these variables. In some cases, it chooses to implement a fuzzy proportional and derivative controller to decrease the overshooting and the period of time, and a fuzzy integral controller is used as a switch for selecting the best response signal. The control the speed of motors using PID and T1 SFLS PID controllers is presented in [3]. The usage of a T1 SFLS PID controller based on two fuzzy logic controllers (FLC) acting as inputs, wher the PID gains are calculated using the Ziegler-Nichols [4], [5]. It can use the simulation to make comparisons between a classic PID controller and a T1 SFLS PID controller, and also evaluate the results of both controllers and observe the differences between them, the differences that often occur between these systems are better signal response, higher reaction rate and thereby the system performance is improved. A control of a single process that uses a T1 SFLS PID and a type-2 SFLS PID is presented by [6]. In this case, three different PID controllers are obtained using a genetic algorithm (GA), named linear PID controller, T1 SFLS PID, and type-2 SFLS PID.

The type-2 SFLS PID offers the best control for the application. The work done in [7] proposes a type-2 FLC to control the position of an actuator in order of few milimeters, and uses the parameters with uncertainty. In [8] an interval type-2 SFLS PID controller with two inputs and one output is used for switching control.

This paper presents a T1 SFLS P-PID controller that online updates the rule base of each gain using the BP algorithm. For the best knowledge of the authors there is not publication reporting the proposed mechanism.

The paper is organized as follow. Section 2 gives the foundations of PID controllers. Section 3 explains the proposed methodology. Section 4 presents the results of the test experiments, and the Section 5 summarizes the conclusions.

II. THEORY (PROFESSIONAL PID, FUZZY LOGIC TYPE-1, BACK PROPAGATION)

2.1 Professional PID

The Professional PID algorithm is able to work in a very reliable manner even when undesirable conditions occur. This thanks to the use of any information that is omitted. Such information can influence the good or bad behavior of this.

The Equation (1) shows the mathematical representation of the professional PID version, which has already been implemented in different jobs with excellent results[9].

$$u(t) = \frac{Ru}{Bp} * \left[e_n(t) - \frac{Td\ \Delta y(t)}{Tc\ R_y} \right] + Int(t) \quad (1)$$

Where: Ru is the actuator range, Bp is the proportional band, $e_n(t)$ is the normalized error, Td is the derivative time, $\Delta y(t)$ is the increase output of the process, Tc is the control period, R_y is the transmitter range with which the variable is measured in physical units, $Int(t)$ represents the integral mode in the PID algorithm.

The Equation (2) shows how to calculate the normalized error, which it is a cornerstone that is implemented in (1).

$$e_n(t) = \frac{y_{rf}(t) - y_f(t)}{R_y} \quad (2)$$

Where $y_{rf}(t)$ is the reference value, which is optionally filtered by a first order exponential filter, $y_f(t)$ is the filtered output value. Int (t) represents the integral mode of the PID algorithm. The calculation of this variable is performed using (3).

$$Int(t) = Int(t-1) + \frac{TcRu}{TiBp} e_n(t) \quad (3)$$

Where $Int(t)$ represents the integral mode in the PID algorithm, $Int(t-1)$ is the previous integral mode of the PID, Tc It is the control period, Ru is the actuator range, Bp is the proportional band, e_n (is the normalized error), Ti is the integral time.

The Equation (4) represents the mathematical model of a first order plant with one delay.

$$y(t) = k_1 * u(t-p) + \alpha * y(t-1) \quad (4)$$

Where: $k_1 = \frac{KTs}{T_1}$, $u(t-p)$ is the signal delay, $\alpha = e^{-\frac{Ts}{T1}}$, $y(t-1)$ is the control signal with one delay.

2.2 Type-1 Fuzzy Logic Systems

The Equation (5) shows the mathematical representation for the fuzzifier of the T1 SFLS system.

$$\mu(k) = \prod_{i=1}^{n} exp\left(-\frac{1}{2}\left(\frac{x_i^* - \bar{x}_i^l}{\sigma_i^l}\right)^2 \right) \quad (5)$$

Where: l is the number of M-rules, i is the number of singleton inputs, $\mu(k)$ is the membership function centered at the measured input $\bar{x}_i^l$ and σ_i^l is the standard deviation.

The Equation (6) shows the mathematical representation for the defuzzifier. With this equation is calculated the output signal of the system.

$$f(x) = \frac{\sum_{l=1}^{m} \bar{y}^l \left(\prod_{i=1}^{n} exp\left(-\frac{1}{2}\left(\frac{x_i^* - \bar{x}_i^l}{\sigma_i^l}\right)^2 \right) \right)}{\sum_{l=1}^{m} \left(\prod_{i=1}^{n} exp\left(-\frac{1}{2}\left(\frac{x_i^* - \bar{x}_i^l}{\sigma_i^l}\right)^2 \right) \right)} \quad (6)$$

The Equation (7) it shows how to combine the gains were entered to PID, where KP_0 is the initial gain of the PID controller and ΔKP is the updated gain calculated by the fuzzy logic. The sum of these two gains is called KP, which is used by the T1 SFLS P-PID controller:

$$KP = \frac{Ru}{Bp} = KP_0 + \Delta KP \quad (7)$$

Likewise, in (8) shows the KI_0 initial gain and the estimated ΔKI:

$$Ti = KI_0 + \Delta KI \quad (8)$$

The initial gain KD_0 and ΔKD to estimate the total derivative gain for KD is represented by (9)

$$Td = KD_0 + \Delta KD \quad (9)$$

2.3 Back Propagation Algorithm

To start using the BP method begins with specifying the structure of the fuzzy system to be implemented. Here we choose the fuzzy system with singleton fuzzifier, center average defuzzyfier, gaussian membership function and the product inference engine [10].

Taking into account these factors, mathematical representation of the type-1 fuzzy system is as (10) in which it is included the above parameters.

$$f(x) = \frac{\sum_{l=1}^{M} y^l \left[\prod_{i=1}^{n} \exp^{\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)} \right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} \exp^{\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)} \right]} \quad (10)$$

Where: x_i is the input of the system, x_i^l is the fuzzy set and σ_i^l is the standard deviation.

To calculate the error of the type-1 fuzzy system the (11) is used:

$$e^p = \frac{1}{2}[f(x_o^p) - y_o^p]^2 \quad (11)$$

The gradient descent algorithm is used to determine the system parameters such as $\bar{y}^l$, $\bar{x}_i^l$, and σ_i^l. The (12) represents the product of the Gaussian membership functions. The input of the system is passed through a product Gaussian operator [10]:

$$z^l = \prod_{i=1}^{n} exp^{\left(-\left(\frac{x_{0i}^p - x_i^l(q)}{\sigma_i^l(q)}\right)^2\right)} \quad (12)$$

Where; x_{0i}^p is the input of the system, $x_i^l(q)$ is the fuzzy set and $\sigma_i^l(q)$ is the standard deviation. The equation (12) is passed through a summation operator and a weighted summation operator to obtain the (13):

$$b = \sum_{l=1}^{m} z^l \quad (13)$$

The Equation (14) represents the sum of the product resulting from the multiplication of $y^l(q)$ by z^l.

$$a = \sum_{l=1}^{m} y^l(q) z^l \quad \textbf{(14)}$$

Where; $y^l(q)$ is the estimator and z^l is the product of all membership function. The output of the fuzzy system is calculated using the (15), where the numerator is (14) and the denominator is (13).

$$f = \frac{a}{b} \quad \textbf{(15)}$$

To determinate $\bar{y}^l$, it is part of the (16).

$$\bar{y}^l(q+1) = y^l(q) - \alpha \frac{\partial e}{\partial y^l} \quad \textbf{(16)}$$

From the (16) the (17) is obtained by the chain rule.

$$\frac{\partial e}{\partial \bar{y}_i^l} = (f\text{-}y)\frac{\partial f}{\partial \alpha}\frac{\partial \alpha}{\partial \bar{y}^l} = (f\text{-}y)\frac{1}{b} z^l \quad \textbf{(17)}$$

Taking both equations defined proceeds to substituting the (17) in the (16) to obtain the (18) which represents the training algorithm for $\bar{y}^l$ for each type-1 fuzzy rule base.

$$\bar{y}^l(q+1) = \bar{y}^l(q) - \alpha \frac{f-y}{b} z^l \quad \textbf{(18)}$$

To determinate $\bar{x}_i^l$ it is part of the (19)

$$\bar{x}_i^l(q+1) = \bar{x}_i^l(q) - \alpha \frac{\partial e}{\partial \bar{x}_i^l} \quad \textbf{(19)}$$

The Equation (20) is obtained from the (19) by the chain rule.

$$\frac{\partial e}{\partial \bar{x}_i^l} = (f\text{-}y)\frac{\partial f}{\partial z^l}\frac{\partial z^l}{\partial \bar{x}_i^l} = (f\text{-}y)\frac{\bar{y}^l\text{-}f}{b} z^l \frac{2\left(x_{0i}^p\text{-}\bar{x}_i^l\right)}{\sigma_i^{l2}} \quad \textbf{(20)}$$

Once obtained the (20) proceeds to perform value substitution presented in (20), these values are substituted in (19) , whereby there is obtained the (21) which represent the training algorithm for $\bar{x}_i^l$ of each fuzzy rule base:

$$\bar{x}_i^l(q+1) = \bar{x}_i^l(q) - \alpha \frac{f-y}{b}\left(\bar{y}^l(q) - f\right) z^l \frac{2\left(x_{0i}^p\text{-}\bar{x}_i^l(q)\right)}{\sigma_i^{l2}(q)} \quad \textbf{(21)}$$

To calculate σ_i^l the same procedure was used to calculate the (22) and (23). That is to say to the 22 that arises is applied the chain rule, the resulting equation is substituted into (22), whereby there is obtained the (23) which represents the training algorithm for σ_i^l .

$$\sigma_i^l(q+1) = \sigma_i^l(q) - \alpha \frac{\partial e}{\partial \sigma_i^l} \quad (22)$$

$$\sigma_i^l(q+1) = \sigma_i^l(q) - \alpha \frac{f-y}{b}\left(\bar{y}^l(q) - f\right) z^l \frac{2\left(x_{0i}^p - \bar{x}_i^l(q)\right)^2}{\sigma_i^{l3}(q)} \quad (23)$$

III. PROPOSED METHODOLOGY

Using Equation (1) [9], type-1 singleton fuzzy logic systems, and the BP algorithm, a fuzzy professional PID controller is proposed. The implementation of each of the three singleton fuzzy systems [10] uses three fuzzy rules previously established for the calculation of ΔKP, ΔKI, and ΔKD, [11].

The design of these fuzzy rules represented in tables 1, 2 and, 3, depends on the process under the control. There are three important variables used to create the tuning of the three controller gains: the error (E), the change of error (EC), and the increment of the gain (ΔK).

The error and the change of error have seven fuzzy sets: NB, NM, NS, Z0, PS, PM, PB are respectively Negative Big, Negative Medium, Negative Small, Zero, Positive Small, Positive Medium, Positive Big.

TABLE 1
FUZZY RULES FOR ΔKD

ΔKD		EC						
		NB	**NM**	**NS**	**Z0**	**PS**	**PM**	**PB**
E	**NB**	PS	NS	NB	NB	NB	NM	PS
	NM	PS	NS	NB	NM	NM	NS	Z0
	NS	Z0	NS	NM	NM	NS	NS	Z0
	Z0	Z0	NS	NS	NS	NS	NS	Z0
	PS	Z0	Z0	Z0	Z0	Z0	Z0	Z0
	PM	PB	NS	PS	PS	PS	PS	PB
	PB	PB	PM	PM	PM	PS	PS	PB

TABLE 2
FUZZY RULES FOR ΔKI

ΔKI		EC						
		NB	**NM**	**NS**	**Z0**	**PS**	**PM**	**PB**
E	**NB**	NB	NB	NM	NM	NS	Z0	Z0
	NM	NB	NB	NM	NS	NS	Z0	Z0
	NS	NB	NM	NS	NS	Z0	PS	PS
	Z0	NM	NM	NS	Z0	PS	PM	PM
	PS	NM	NS	Z0	PS	PS	PM	PB
	PM	Z0	Z0	PS	PS	PM	PB	PB
	PB	Z0	Z0	PS	PM	PM	PB	PB

TABLE 3
FUZZY RULES FOR ΔKP

ΔKP		EC						
		NB	**NM**	**NS**	**Z0**	**PS**	**PM**	**PB**
E	**NB**	PB	PB	PM	PM	PS	Z0	Z0
	NM	PB	PB	PM	PS	PS	Z0	NS
	NS	PM	PM	PM	PS	Z0	NS	NS
	Z0	PM	PM	PS	Z0	NS	NM	NM
	PS	PS	PS	Z0	NS	NS	NM	NM
	PM	PS	Z0	NS	NM	NM	NM	NB
	PB	Z0	Z0	NM	NM	NM	NB	NB

Each gain has a different output but the data of error and change of error are the same. Table 1 shows the fuzzy rules designed for the ΔKP gain estimation. The Table 2 shows the fuzzy rules designed for the ΔKI gain. Table 3 shows the fuzzy rules designed for the ΔKD gain. Each fuzzy rule base has an array of 49 fuzzy rules for the output signal, ΔKP, ΔKI, ΔKD respectively. Based on the results of tests it was decided to assign high values to the output signal in the table designed for ΔKP, the values assigned to control this gain ranging from 40 to 150. In the case of the ΔKI gain the selected values are in the range of 0 to 0.15. Finally, the gain ΔKD receives values from -0.3 to +0.3 choosing between these ranges only seven values for each of the gains with which have been working.

The online tuning is a critical part of this proposal because the values of the fuzzy sets of each rule is tuned using the ideal values that the increment of each gain minimizes the error in the control signal and maintains the stability of the response of the plant, eliminating the overshooting on the behavior of the plant.

IV. RESULTS

In this section, we show the results obtained with the implementation of the proposed T1 SFLS P-PID controller with the BP training. The steady-state error is a measure of the accuracy and performance of the proposed controller. Fig. 1 shows the basic structure of the T1 SFLS P-PID-BP controller. The learning criterion used for the BP algorithm is $E_u = 0, \ \Delta E_u = 0$ This criterion is used for the online tunning of each fuzzy rule base.

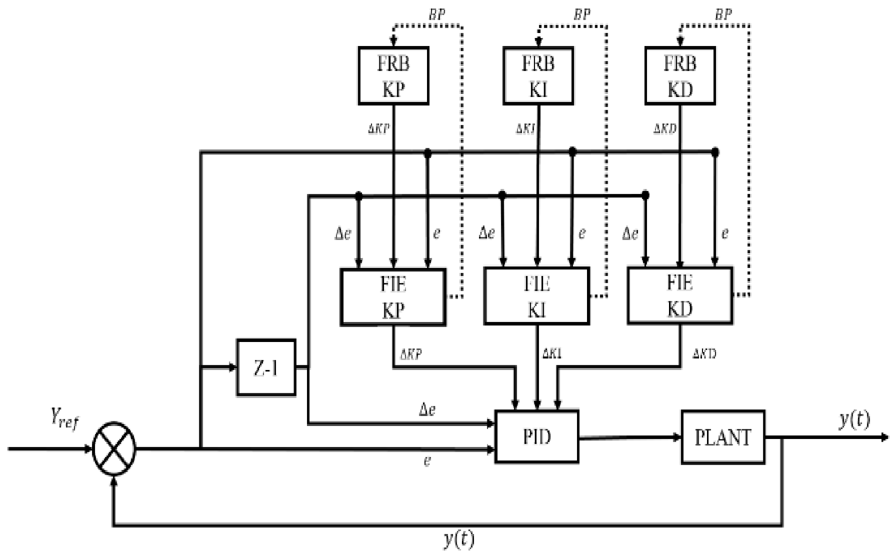

FIG. 1 BASIC STRUCTURE OF T1 SFLS P-PID-BP (FIE: FUZZY INFERENCE ENGINE, FRB: FUZZY RULE BASE)

The Fig. 2 shows the response of the professional PID controller. As observed, occurs an overrun set by set-point, this signal is stable and the response time of the signal is faster than the traditional PID.

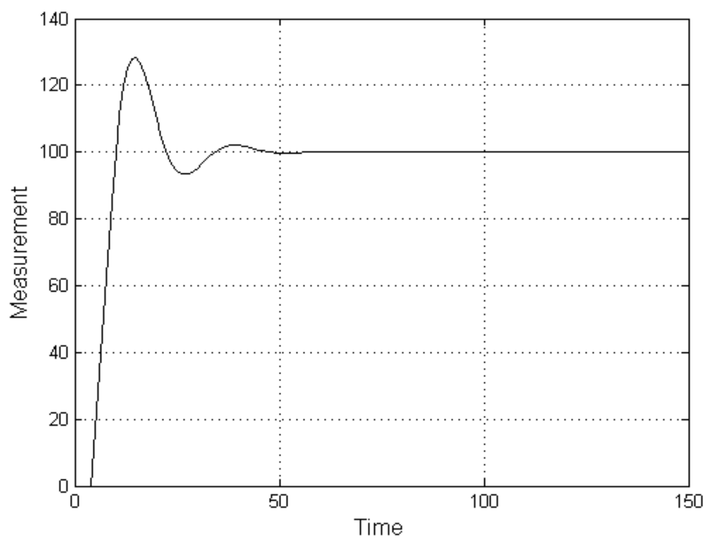

FIG. 2 PROFESSIONAL PID

Fig.3 shows the professional PID singleton type-1 which apply fuzzy logic behavior, and as can be seen, an improvement is obtained in the output response. The fuzzy logic implemented in this case use the fuzzy rules generated for the gains of KP, KI and KD.

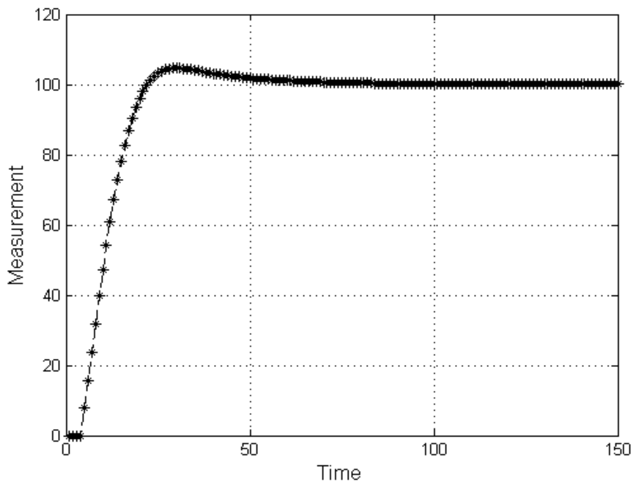

FIG. 3 T1 SFLS P-PID

Fig. 4 shows the response signal behavior of the professional PID singleton type-1 using the BP method for the online update of the fuzzy rules.

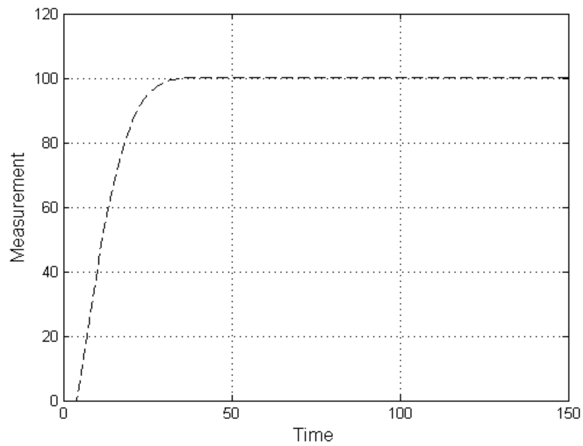

FIG. 4 T1 SFLS P-PID-BP

Fig. 5 shows a comparison between a professional PID, T1 SFLS P-PID and T1 SFLS P-PID-BP controllers. As observed in Fig. 5 the professional T1 SFLS P-PID represented by the line (*) has a better response to the output in comparison with the professional PID represented by line (.), because a signal overrun occurs set by set-point less than that which occur in the professional PID, however this response time is less than the professional PID.

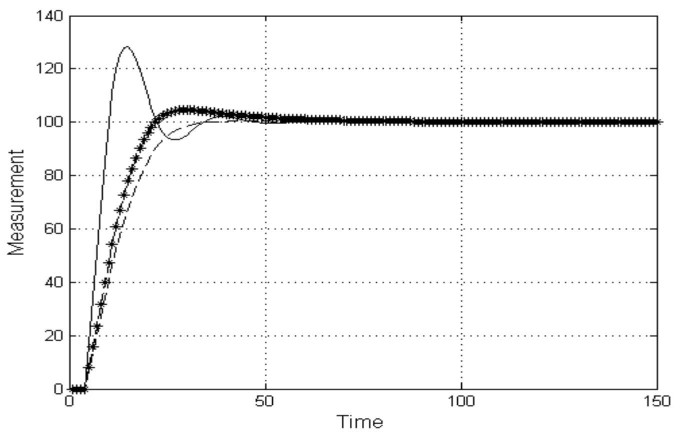

FIG. 4 COMPARISON BETWEEN THE CONTROLLERS: PROFESSIONAL PID (.), T1 SFLS P-PID (*), T1 SFLS P PID-BP (--).

Now the T1 SFLS P-PID-BP represented by the line (--) shows better response to the output with a shorter response time. Table 4 shows the values obtained from the behavior of the three controllers: a) the maximum overshooting, b) the time in which the maximum overshooting is presented, and c) the time at which the system stabilizes.

TABLE 4
RESULTS OF THE TIME FOR STABILIZATION, MAXIMUN OVERSHOOTING, AND THE TIME FOR THE MAXIMUM OVERSHOOTING.

Type of controller	Max. over-shooting	Time to max overs-hooting	Time for stabilization
PID	128.3	15	79
FPID	104.6	29	110
FPID-BP	100.3	44	93

V. CONCLUSIONS

According to the experimental results, the T1 SFLS P-PID-BP presents the better performance in comparison with the two benchmarking controllers, the P-PID and the T1 SFLS P-PID controllers. This was achieved because of the implementation of the mechanism to update the fuzzy rules. The overshooting of the behavior of the controlled plant is eliminated, and the velocity of the response is faster.

REFERENCES

[1] Z. Shi, T. Wang, D. Liu, C. Ma, and X. Yuan, "A fuzzy PID-controlled sma actuator for a two-dof joint," *Chinese J. Aeronaut.*, vol. 27, no. 2, pp. 453–460, 2014.
[2] J. Xu and Z. Feng, "A novel self-adaptive fuzzy-PID controller for temperature control in variable refrigerant volume (VRV) air conditioning systems," *Proc. Intell. Syst. Knowl. Eng.*, pp. 1–5, 2007.
[3] C. Igleheart, J. Marques, C. Ramirez-Leon, Y. Li, F. Ashrafzadeh, and S. Wilson, "A robust PID-type fuzzy logic controller for variable speed motor drives," *45th North American Power Symposium, NAPS 2013*, kentucky, 2013.
[4] R. Arulmozhiyal, "Design and implementation of fuzzy PID controller for brushless dc motor using fpga," in *IEEE,International Conference on Power Electronics, Drives and Energy Systems*, 2012.
[5] R. Arulmozhiyal and R. Kandiban, "Design of fuzzy PID controller for brushless dc motor," *2012 Int. Conf. Comput. Commun. Informatics*, pp. 1–7, 2012.
[6] H. Araujo, B. Xiao, C. Liu, Y. Zhao, and H. K. Lam, "Design of type-1 and interval type-2 fuzzy PID control for anesthesia using genetic algorithms," no. May, pp. 70–93, 2014.
[7] G. Kothapalli and M. Y. Hassan, "Fuzzy controlled hydraulic excavator with model parameter uncertainty," *World Acad. Sci. Eng. Technol.*, vol. 60, pp. 1889–1894, 2011.
[8] Y. Chen and T. Wang, "Interval type-2 fuzzy PID control and simulation," *2nd Int. Conf. Electron. Mech. Eng. Inf. Technol.*, pp. 326–330, 2012.
[9] A. Aguada, "Temas de Identificación y Control Adaptable," La Habana, Cuba, Instituto de Cibernética, Matemáticas y Física, 2000.
[10] L.-X. Wang, *A course in fuzzy systems and control*. Pretnice Hall.
[11] D. Shi, G. Gao, Z. Gao, and P. Xiao, "Application of expert fuzzy PID method for temperature control of heating furnace," *Procedia Eng.*, vol. 29, pp. 257–261, 2012.