PID Implemented by a Type-1 Fuzzy Logic System with Back-Propagation Algorithm for Online Tuning of Its Gains

Alberto Álvarez, David Reyes, Ernesto J. Rincón, José Valderrama,
Pascual Noradino, and Gerardo M. Méndez(✉)

Instituto Tecnológico de Nuevo León, Av. Eloy Cavazos # 2001, Col. Tolteca,
Cd., 67170 Guadalupe, NL, Mexico
gmm_paper@yahoo.com.mx

Abstract. Two different types of benchmarking proportional-integral-derivative PID controllers are used to compare the proposed methodology. In the first controller 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 Ziegler Nichols method. The second controller uses three type-1 fuzzy logic systems to estimate each one of the gains every control cycle. This paper proposes a fuzzy self-tuning 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 T1 SFLS PID-BP. The results show that the proposed controller presents better performance than the two benchmarking controllers: the PID and the T1 SFLS PID.

1 Introduction

The first contribution of this proposal is the online update of the gains for a PID controller which is a fundamental part due the values of the fuzzy sets of each rule are tuned using ideal values through the increase of the variables of each gain which minimize the error signal of control and still the stability for the plant response, many researches were made for improve the signal of control, between them can be found some as mentioned below.

In [1] a fuzzy controller is used to reduce the overstress that arise in the plant so as improve 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. The usage of a T1 SFLS PID controller based on two fuzzy logic controllers (FLC) acting as inputs, where the PID gains are calculated using the Ziegler-Nichols are presented in both [3, 4]. 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. A control of a single process that uses a T1 SFLS PID and a type-2 SFLS PID is presented by [5]. In this case, three different PID controllers are obtained

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_28

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 [6] processes a type-2 FLS to control the position of an actuator in order of few millimeters, and uses the parameters with uncertainty. In [7] an interval type-2 SFLS PID controller with two inputs and one output is used for switching control. Has been presented a hybrid algorithm for interval type-1 non-singleton type-2 TSK fuzzy logic system [8] that is capable of compensate for uncertain measurements, on this algorithm is used the back-propagation (BP) method for calculate the backward pass, the error propagates backward and the antecedent parameters. In [9] is used an Interval type-1 non-singleton type-2 fuzzy logic system for control the strip head end thickness calculating the roll gap however not is presented any method for update the controller gains like Back-propagation. In [10] a fuzzy self-tuning PID Smith is proposed, the error and change of error are the inputs, which enter to the fuzzification, then to fuzzy inference based and after the parameters of PID control are obtained when add these to the gains. A Non-singleton type-1 fuzzy logic system is used like an input that update each gain of a PID controller [11] which is applied on an Atmega 2560 for controlling a stepper motor. In [12] is used a PID control and a fuzzy system for longitudinal control of an air craft where a type-1 fuzzy system and a PID controller are used for comparison.

The present work introduce a novel methodology that update the rule base of a type-1 fuzzy logic system for the gains of a PID controller through an BP algorithm with the purpose of improve the output signal.

2 Mathematical Models

The 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.

The Eq. (1) shows the mathematical representation of the proposed PID version, which has already been implemented in different jobs with excellent results [13].

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

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, Ry is the transmitter range with which the variable is measured in physical units, $Int(t)$ represents the integral mode in the PID algorithm. The Eq. (2) shows how to calculate the normalized error for (1) where $y_{rf}(t)$ is the reference value, $y_f(t)$ is the filtered output value. The calculation for $Int(t)$ variable is performed using (3) where $Int(t-1)$ is the previous integral mode of the PID, Tc is the control period, Ti is the integral time. The Eq. (4) represents the mathematical model of a first order plant with one delay.

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

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

$$y(t) = k_1 u(t-p) + \alpha y(t-1) \tag{4}$$

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

Type-1 FLS (5) shows the mathematical representation for the fuzzifier of the T1 SFLS 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 x_i^{-l} and σ_i^l is the standard derivation. The equation for the deffuzifier is shown in (6) with this equation is calculated the output signal of the system.

$$\mu(k) = \prod\nolimits_{i=1}^{n} \exp\left(-\frac{1}{2}\left(\frac{x_i^* - x_i^{-l}}{\sigma_i^l}\right)^2\right) \tag{5}$$

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

Equations (7, 8 and 9) show 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 FLS, likewise, in (8) KI_0 initial gain and the estimated ΔKI and KD_0 with ΔKD to estimate the total derivative gain for KD is represented by (9).

$$KP = \frac{Ru}{Bp} = KP_0 + \Delta KP \tag{7}$$

$$Ti = KI_0 + \Delta KI \tag{8}$$

$$Td = KD_0 + \Delta KD \tag{9}$$

To start using the *BP* method begins with specifying the structure of the FLS to be implemented. Here we choose the FLS with singleton fuzzifier, center average defuzzyfier, Gaussian membership function (GMF) and the product inference engine [14]. Taking into account these factors, mathematical representation of the type-1 FLS is as (10) in which it is included the above parameters 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 FLS the (11) is used:

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

$$e^p = \frac{1}{2}\left[f\left(x_0^p\right) - y_0^p\right]^2 \tag{11}$$

The gradient descent algorithm is used to determine the system parameters such as y^{-l}, x_i^{-l}, and σ_i^l. (12) is the product of the GMF. The input of the system is passed through a product GMF operator [9] where: x_{oi}^p is the input of the system $x_i^l(q)$ is the fuzzy set and $\sigma_i^l(q)$ is the standard deviation. (12) is passed through a summation operator and a weighted summation operator to obtain the (13):

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

$$b = \sum_{l=1}^{m} z^l \tag{13}$$

The Eq. (14) represents the sum of the product resulting from the multiplication of $y^l(q)$ by z^l where: $y^l(q)$ is the estimator and z^l is the product of all membership functions. The output of the fuzzy system is calculated using the (15), where the numerator is (14) and the denominator is (13).

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

$$f = \frac{a}{b} \tag{15}$$

To determinate y^{-l}, it is part of the (16). From the (16) the (17) is obtained by the chain rule.

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

$$\frac{\partial e}{\partial y_i^{-l}} = (f - y)\frac{\partial f}{\partial \alpha}\frac{\partial \alpha}{\partial y^{-l}} = (f - y)\frac{1}{b}z^l \tag{17}$$

Then proceeds to substituting the (17) in the (16) to obtain the (18) which represents the training algorithm for y^{-l} for each type-1 fuzzy rule base. To determinate x_i^{-l} it is part of the (19). (20) is obtained from the (19) by the chain rule.

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

$$x_i^{-l}(q+1) = x_i^{-l}(q) - \alpha \frac{\partial e}{\partial x_i^{-l}} \tag{19}$$

$$\frac{\partial e}{\partial x_i^{-l}} = (f-y)\frac{\partial f}{\partial z^{l}}\frac{\partial z^{l}}{\partial x_i^{-l}} = (f-y)\frac{y^{-l}-f}{b} z^{l^2}\frac{\left(x_{0i}^{p} - x_i^{-l}\right)}{\sigma_i^{l^2}} \tag{20}$$

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 x_i^{-l} of each fuzzy rule base:

$$x_i^{-l}(q+1) = x_i^{-l}(q) - \alpha \frac{f-y}{b}\left(y^{-l}(q) - f\right) z^{l^2}\frac{\left(x_{0i}^{p} - x_i^{-l}(q)\right)}{\sigma_i^{l^2}(q)} \tag{21}$$

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.

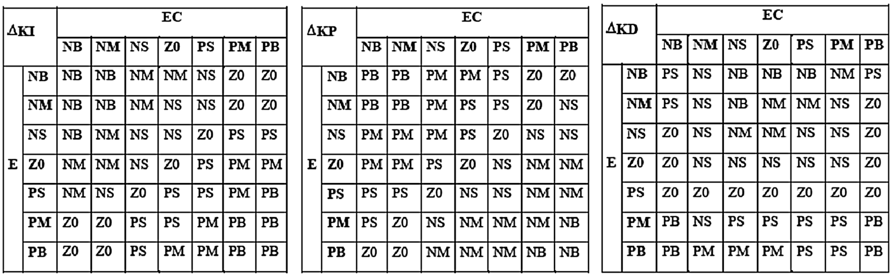
$$\sigma_i^{l}(q+1) = \sigma_i^{l}(q) - \alpha \frac{\partial e}{\partial \sigma_i^{l}} \tag{22}$$

$$\sigma_i^{l}(q+1) = \sigma_i^{l}(q) - \alpha \frac{f-y}{b}\left(y^{-l}(q) - f\right) z^{l^2}\frac{\left(x_{0i}^{p} - x_i^{-l}(q)\right)^2}{\sigma_i^{l^3}(q)} \tag{23}$$

3 Proposed Methodology

Using (1) [13], type-1 SFLS, and the *BP* algorithm, aFL PID controller is proposed. The implementation of each of the three SFLS [14] uses three fuzzy rules previously established for the calculation of Δ*KP*, Δ*KI* and Δ*KD*, [15]. The design of these fuzzy rules represented in Table 1 depends on the process under the control. These 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 (Δ*K*). The error and the change of error have seven fuzzy sets: NB, NM, NS, ZO, PS, PM and PB are respectively Negative Big, Negative Medium, Negative Small, Zero, Positive Small, Positive Medium, Positive Big. 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 Δ*KP* gain estimation, the fuzzy rules designed for the Δ*KI* gain and the fuzzy rules designed for the Δ*KD* gain. Each fuzzy rule base has an array of 49 fuzzy rules for the output signal Δ*KP*, Δ*KI* and Δ*KD* respectively.

Table 1. (a) Fuzzy Rules for Δ*KI*, (b) Fuzzy Rules for Δ*KP*, (c) Fuzzy Rules for Δ*KD*.



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

(a)

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

(b)

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

(c)

4 Results

Were trained three different controllers for compare the response of its signals with the final purpose of see the performance of the proposed methodology.

In this section, the results are shown with the implementation of the proposed T1 SFLS PID controller with the BP training where it can see the done iteration for the BP algorithm between the fuzzy inference engine (FIE) and the fuzzy rule bases (FRB), here basically the BP update the FRB which in turn update the FIE and the same procedure is done in the three gains KP, KI and KD for improve the response of the PID signal. Figure 1 shows the basic structure of the T1 SFLS PID-BP controller. The learning criterion used for the BP algorithm is $E_u = 0$, $\Delta E_u = 0$. The steady-state error (Fig. 2c) is a measure of the accuracy and performance of the proposed controller.

In the Fig. 2 the pointed line shows the response of the PID controller. As observed, occurs an overrun set by set-point. Intermittent line shows the 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

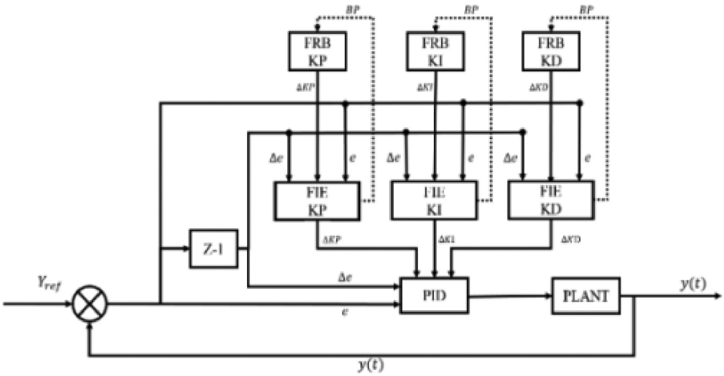


Fig. 1. Structure of T1 SFLS PID-BP.

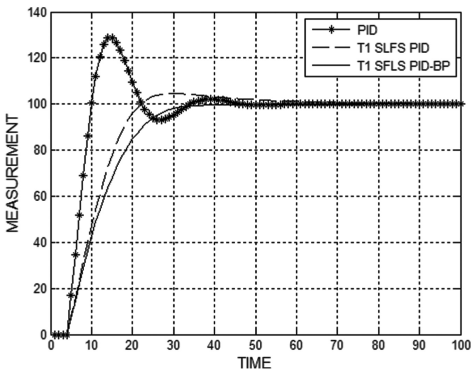


Fig. 2. Comparison between (a) PID, (b) T1 SFLS PID and (c) T1 SFLS PID-BP

rules generated for the gains of KP, KI and KD. And, continues line shows the response signal behavior of the PID singleton type-1 using the BP method for the online update of the fuzzy rules.

As observed in Fig. 2 the T1 SFLS PID represented by the intermittent line has a better response to the output in comparison with the PID represented by the pointed line, due the signal in T1 SFLS PID has a more low overshooting, however the response in the time and the stabilization are more slow than the PID. Now the T1 SFLS PID-BP represented for continue line shows a response in the output with the most minimal overshooting, an average stabilization and with a bad point in the maximum time in the overshooting. Table 2 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 in which the system stabilizes.

Table 2. Time for stabilization max overshooting, and the time for the max overshooting

Type of Controller	Maximum over-shooting	Time to max. over-shooting	Time for stabilization
PID	128.3	15	79
FPID	104.6	29	110
FPID-BP	100.3	44	93

5 Conclusions

According to the experimental results, the T1 SFLS PID-BP presents the better performance in comparison with the two benchmarking controllers due that have a smoother signal of response that the PID and the T1 SFLS PID controllers. This was achieved due the implementation of the mechanism to update the fuzzy rules. Further, the overshooting for the behavior of the plant is reduced until 0.3 which is much better for the control of actuators in T1 SFLS PID-BP, and the velocity of the response is faster. Although the speed is not very good in comparison with the others controller the stability in stationary state is better because it present a smaller damping.

References

1. Shi, Z., Wang, T., Liu, W., Ma, C., Yuan, X.: A fuzzy PID-controlled SMA actuator for a two-DOF joint. Chinese J. Aeronaut. **27**(2), 453–560 (2014)
2. Xu, J., Feng, Z.: A novel self-adaptive fuzzy-PID controller for temperature control in variable refrigerant volume (VRV) air conditioning systems. In: Proceedings of the Intelligent System Knowledge Engineering, pp. 1–5 (2007)
3. Arulmozhiyal, R.: 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)
4. Azulmozhiyal, R., Kandiban, R.: Design of fuzzy PID controller for brushless DC motor. In: 2012 International Conference on Computer Communication and Informatics, pp. 1–7 (2012)
5. Araujo, H., Xiao, B., Liu, C., Zhao, Y., Lam, H.K.: Design of type-1 and interval type-2 fuzzy PID control for anesthesia using genetic algorithms, 70–93 (2014)
6. Kothapalli, G., Hassan, M.Y.: Fuzzy controlled hydraulic excavator with model parameter uncertainty. World Acad. Sci. Eng. Technol. **60**, 1889–1894 (2011)
7. Chen, Y., Wang, T.: Interval type-2 fuzzy PID control and simulation. In: 2nd International Conference on Electronic & Mechanical Engineering and Information Technology, pp. 326–330 (2012)
8. Mendez, G.M., Hernandez, M.A.: Interval type-1 non-singleton type-2 TSK fuzzy logic systems using the hybrid training method RLS-BP. Analysis and Design of Intelligent Systems using Soft Computing Techniques, vol. 41, pp. 36–44 (2007)
9. Mendez, G.M., Castillo, O., Colás, R., Moreno, H.: Finishing mill strip gage setup and control by interval type-1 non-singleton type-2 fuzzy logic systems. Appl. Soft Comput. **24**, 900–911 (2014)
10. Jie, S., Dian-hua, Z., Xu, L., Jin, Z.: Smith prediction monitor AGC system based on fuzzy self-tuning PID control. J. Iron Steel Res. **17**(2), 22–26 (2010)
11. Ramos, J.M.J., Reyes, E., Sanchez, J.L., Mendez, G.M.: A professional PID implemented using a non-singleton type-1 fuzzy logic system to control a stepper motor. Int. J. Eng. Sci. **2** (2), 94–101 (2016)
12. Castillo, O., Cervantes, L.: Genetic design of optimal type-1 and type-2 fuzzy systems for longitudinal control of an airplane. Intell. Autom. Soft Comput. **20**(2), 213–227 (2014)
13. Aguado, A.: Temas de Identificación y Control Adaptable. Instituto de Cibernética, Matemáticas y Física, La Habana (2000)
14. Wang, L.-X.: A Course in Fuzzy Systems and Control. Pretince Hall, Upper Saddle River
15. Shi, D., Gao, G., Xiao, P.: Application of expert fuzzy PID method for temperature control of heating furnace. Procedia Eng. **29**, 257–261 (2012)