Real-time Environment for the Design and Evaluation of Fuzzy Controllers

Fernando Cofman[†], Doris Sáez[‡] and Juan C. Gómez[§]

[†] Electronics Department, Faculty of Engineering, University of Buenos Aires, Argentina
fcofman@fi.uba.ar
[‡] Electrical Engineering, University of Chile, Chile
dsaez@ing.uchile.cl
[§] Informatics Technical Unit (UTI), Telecommunications, Electronics and Informatics Research Centre (CITEI), National Institute of Industrial Technology (INTI), Argentina
juanca@inti.gov.ar

Abstract. This paper presents the results of the implementation of Fuzzy PID control strategies on a complex, critically-stable system, namely the PID-Pong machine. Two different fuzzy algorithms are implemented in real-time and compared to a linear PID controller, used as the evaluation basis. Conclusions are established based on regulation, implementation and tuning indicators.

1 Introduction

The use of Fuzzy PID controllers has raised notorious interest in the industry in the last years. The reasons can be summed up into: similarity to the well-known PID controllers and capabilities to compensate for non-linearities in a wide variety of systems. However, there are a number of setbacks while implementing these novel algorithms.

As stated by Li *et al* (1995), the most difficult problem with Fuzzy Logic Controllers is the parameter tuning; the scaling gains for such controllers are adjusted through trial and error due to the lack of systematic methods. Admitting that it is difficult to maintain excellent performance in both, transient and steady state, a gain-scheduling solution is presented in Li's work. The approach aims to determine two sets of gains: one aimed to enhance the transient response and the other to achieve a smoothed steady state. Despite the improved performance obtained, the method to determine such gains is quite informal: "... tune *KU* to have reasonable performance ...", "... tune *KE*, *KCE* to improve performance ...".

Ying *et al* (1990) and Li *et al* (1996) have contributed to this, by reducing the number of parameters to be adjusted. This is accomplished by finding a relation between the original gains or, in other words, assuming that there exists a proportionality factor that relates gains *KE* and *KCE*. Choi *et al* (2000) proposes further simplifications by reducing the number of inputs to the controller by means of a clever infinitesimal approximation. Nevertheless, all these works still cannot provide a systematic method for tuning the parameters.

Other authors have introduced useful guidelines to provide additional knowledge regarding the influence of the different gain factors on the closed-loop system's response. Such are the cases of Tancara (1993) and Li (1997), who explain equivalences in the parameters' adjustments in the form: "... An increase in (certain fuzzy parameter) has the same effect than an increase/decrease in (certain linear parameter) ...". Helpful, though qualitative, these guides yet fail to achieve a systematic method. Instead, they can only be used as common-sense recommendations, as in linear control.

To overcome these shortcomings, some 'intelligent' controllers have been presented. Among others, there can be counted adaptive controllers (Victor *et al*, 1997), auto-tuning controllers (Escamilla *et al*, 2002), variable structure controllers (Li *et al*, 1997) and even optimized controllers, using genetic algorithms (Hu *et al*, 1999) and neural networks (Lee *et al*, 2001). The latter two, do provide a complete systematic method for tuning all of the controller's parameters, however they require deep a-priori knowledge of the system, which is not always available. In any case, none of them is considered in this work.

This paper is divided in five sections, of which the first one is an introduction. Section 2 describes in detail the experimental system used for the real-time tests. In section 3 a discussion of the different controllers, the theory behind them and the tuning maxims used in each of the cases (except for the linear case, which is assumed to be widely known) is presented. All of the experimental tests, including the evaluation basis, the empirical results and their comparison, are presented in section 4. Finally, section 5 points out the conclusions of this paper.

2 The PID-Pong Machine (experimental system)

The PID-Pong machine was proposed by Cantrell (1994) and consists of a plastic sphere constrained to a transparent acrylic tube (3.5 *ft* long by 1 *inch* diameter) with a DC fan at its base, which keeps the ball in levitation, as shown in Figure 1.

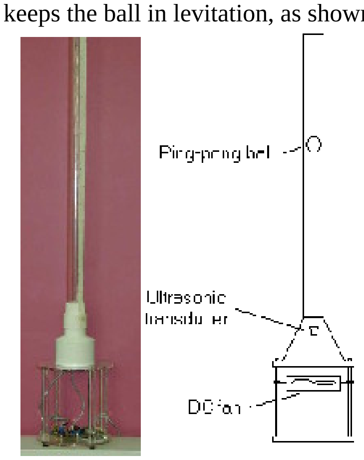

Fig. 1. PID-Pong machine (real system and schematic diagram)

An ultrasonic transducer placed just above the fan is used to measure the ball's height and a pulse-modulated signal applied to the motor regulates its rotation speed.

The system exhibits critically-stable behavior, open-loop integral action and a very high sensitivity to perturbations, these being the reasons for considering it an interesting challenge for control.

The control objective is to stabilize the ball's height by means of the signal applied to the fan.

The closed-loop system architecture consists of an electronic interface, fully micro-processed, which commands all the actions performed on the plant. It also establishes a serial link to a terminal, where the controller's core resides, providing the feedback signal.

The interface has been designed and built with solid-state electronic components, ensuring proper stability and repeatability of the tests. The *ad-hoc* protocol transmits every 0.1 *s* the result of the height measurement and accepts a 1-byte-long command, representing the pulse modulation setting, calculated by the controller running on the terminal side. The controller has been implemented using MATLAB©, Simulink© and RTW©; a custom serial driver has been designed and programmed in C for this application. The information received from the interface is decimated to 0.2 *s*, pre-filtered with a 10th order moving-average filter and logged into MATLAB© compatible files for off-line analysis.

3 Fuzzy Controllers Design

3.1 Fuzzy PI

The controller described in this section is derived form the incremental approach, proposed by Lee (1990). It features two inputs (the error and its increment) and one output (the increment in the manipulated variable).

$$Controller_{Inputs}: \quad e(k) = r(k) - y(k) \quad , \quad de(k) = e(k) - e(k-1) \tag{1}$$

$$\begin{aligned} Controller_{Output}: &\quad du(k) = u(k) - u(k-1) \\ \Rightarrow &\quad u(k) = \sum_{i=0}^{k} du(i) \end{aligned} \tag{2}$$

The definition of the system error as a function of time *e*(*k*), is the difference between the observed variable *y*(*k*) (i.e. the ball height) and its setting value or desired position *r*(*k*), as shown in Equation 1. Also in Equation (1), the definition of the error increment *de*(*k*) is presented; it consists of the difference between two time-successive error values: current and previous. Finally, Equation (2) introduces the controller output, again as the difference between two time-successive values, but in this case of the manipulated variable *u*(*k*) (i.e. the fan pulse modulation). Note that the actual manipulated variable applied to the experimental system results in the accumulation of the controller output.

Each of these variables is affected by a scaling factor (gain) to be tuned: *KE*, *KCE* and *KCU*, respectively, which have to be carefully chosen to avoid saturation in the fuzzification process.

The fuzzy sets defined for the variables consist of ‘n’ triangular membership functions, evenly distributed along the scaled inputs and output ranges. The rules of the fuzzy controller are derived from the phase plane analysis, as explained by Lee. They can be summarized as shown in Table 1, where the case of 7 input fuzzy sets and 7 output fuzzy sets is presented.

		***de*(*k*)**						
		NB	NM	NS	ZR	PS	PM	PB
***e*(*k*)**	PB	ZR	PS	PM	PB	PB	PB	PB
	PM	NS	ZR	PS	PM	PM	PB	PB
	PS	NM	NS	ZR	PS	PS	PM	PB
	ZR	NM	NM	NS	ZR	PS	PM	PM
	NS	NB	NM	NS	NS	ZR	PS	PM
	NM	NB	NB	NM	NM	NS	ZR	PS
	NB	NB	NB	NB	NB	NM	NS	ZR

Table 1. Rules for a controller with 7 fuzzy sets; *du*(*k*) as a function of *e*(*k*) and *de*(*k*).

The decision logic for the inference process is based on the min operator for implication and the MOM (Mean Of Max) method for de-fuzzification.

The tuning procedure resides in finding the appropriate value for *KE*, *KCE* and *KCU*. As shown by Ying (1990), it is possible to find a parallelism between *KE*, *KCE* and *KCU* and the well known *KP* and *KI* of a PID controller.

With this starting point, the fine-tuning is performed through trial and error, following Tancara’s (1993) and Li’s (1997) guidelines.

$$KP = \frac{KCU.KCE}{4} \tag{3}$$

$$KI = \frac{KCU.KE}{4} \tag{4}$$

3.2 Hybrid Fuzzy PID

In this section, another Fuzzy PID controller is presented (Escamilla *et al*, 2002), featuring hybrid architecture with a Takagi-Sugeno (1985) structure for the consequences. Its principal advantages are that the response can be made to match that of the linear PID (this can be used as a starting point for tuning the fuzzy controller when the linear approach achieves stable response but loose – inadequate – regulation characteristics), and that it is possible to modify the controller's structure to better compensate for the process's behavior.

As in the previous case, the inputs to the controller are the error and its increment, as presented in Equation (1). For the output, a slight modification is considered. For instance, the fuzzy inference system calculates a single variable, which will be referred to as *U*(*k*). This new variable is used as both, the output and its increment of the controller. Thus, the actual manipulated variable is calculated considering *U*(*k*) and its accumulated value, as shown in Equation (5), as opposed to the previous PI controller in Equation (2).

Similarly, gain scaling factors are defined for each of the variables: *GE*, *GCE*, *GU* and *GCU*. Note that in Equation (5), these gains have been included.

$$u(k) = GU.U(k) + GCU.\sum_{i=0}^{k} U(i) \tag{5}$$

The fuzzy sets defined for the input variables are, again, triangular. Only two sets are used, so that for any value of the input variables, the membership degree of both sets adds 1 (complementary property). Given that the fuzzy system is based on Takagi-Sugeno models, there are no fuzzy sets defined over the outputs. Instead, a crisp linear function is used for the consequence of each rule.

All the possible combinations of antecedents determine the complete set of rules, thus:

$$\begin{aligned} R_1:\ & \mathit{if} \quad (e \ \ \mathit{is} \ \ -) \ \ \mathit{and} \ \ (de \ \ \mathit{is} \ \ -) \\ & \mathit{then} \quad U_1(k) = p_1.e(k) + q_1.de(k) + c_1 = e(k) + de(k) \end{aligned} \tag{6}$$

$$\begin{aligned} R_2:\ & \mathit{if} \quad (e \ \ \mathit{is} \ \ -) \ \ \mathit{and} \ \ (de \ \ \mathit{is} \ \ +) \\ & \mathit{then} \quad U_2(k) = p_2.e(k) + q_2.de(k) + c_2 = e(k) + de(k) \end{aligned} \tag{7}$$

$$\begin{aligned} R_3:\ & \mathit{if} \quad (e \ \ \mathit{is} \ \ +) \ \ \mathit{and} \ \ (de \ \ \mathit{is} \ \ -) \\ & \mathit{then} \quad U_3(k) = p_3.e(k) + q_3.de(k) + c_3 = e(k) + de(k) \end{aligned} \tag{8}$$

$$\begin{aligned} R_4:\ & \mathit{if} \quad (e \ \ \mathit{is} \ \ +) \ \ \mathit{and} \ \ (de \ \ \mathit{is} \ \ +) \\ & \mathit{then} \quad U_4(k) = p_4.e(k) + q_4.de(k) + c_4 = e(k) + de(k) \end{aligned} \tag{9}$$

The decision logic uses the ? (product) operator for the fuzzy implication and the weighted mean for the de-fuzzification.

As stated before, given the tuned parameters of a classical PID controller (or the Ziegler-Nichols closed-loop method critical constants), the hybrid system response can be made to identically match that of the linear one:

$$GE = 1 \tag{10}$$

$$GU = \frac{KP}{2} = \frac{0.6 \cdot KU}{2} = 0.3 \cdot KU \tag{11}$$

$$GCE = \frac{KP}{T_I} = \frac{0.6 \cdot KU}{0.5 \cdot TU} = 1.2 \cdot \frac{KU}{TU} \tag{12}$$

$$GCU = 2 \cdot \frac{KP \cdot T_D}{KP} = 2 \cdot 0.125 \cdot TU = 0.25 \cdot TU \tag{13}$$

Noting that the consequence's coefficients are arbitrarily set to $p_i = q_i = 1$ and $c_i = 0$, it is possible to better adapt the controller to the plant's behavior, by modifying these parameters accordingly. The flexibility of the structure is here implied.

4 Experimental Tests

4.1 Evaluation Basis

The first group of performance indexes used is related to regulation (i.e. the capabilities of the control system to achieve and maintain the control goal). These are: *IAE* (integral of absolute error), *ITAE* (integral of time absolute error) and *CE* (control effort).

$$IAE = \int |e|.dt \tag{14}$$

$$ITAE = \int t.|e|.dt \tag{15}$$

$$CE = \sum_{T-1} \left(u_k - u_{k-1}\right)^2 \tag{16}$$

The former two indexes measure the ability of the controller to keep the observed variable as close as possible to the reference and were previously used by Tang (1987), Pedrycz (1995), Li (1996 and 1997) and others. Particularly, the first one emphasizes the error during the transient, while the second one accounts mainly for the steady state. Finally, *CE* shows the ability of the controller to provide a smooth response to a given change in setting or a perturbation.

The second group of indexes is related to the effort needed to implement each of the control algorithms. These include: *SF* (structure flexibility), *IE* (implementation effort), *TE* (tuning effort) and *FTE* (fine-tuning effort).

As opposed to the first group of indexes, these parameters are defined in a somewhat more subjective way. Nevertheless, they are important and have to be considered when the decision on whether one alternative or the other is to be implemented (especially in big scale enterprises).

The *SF* coefficient shows the ability of the controller to compensate for different non-linearities in a system. The *IE* is based on the mathematical operations needed to obtain the controller's output. It is calculated adding the number of Simulink© blocks used, multiplied by their weighting factor, as it can be seen in Table 2. Particularly, the fuzzy rule implication is composed of two Min/Max blocks, thus its weighting factor is twice as much the latter's one. Also the Discrete Filter is composed of Sum and Product operations; nevertheless, a 10th order filter block eliminates much of the function-call overhead compared to a series of 10 Sum and Product blocks, resulting in half as much the weighting factor expected for the series.

Simulink© block	**Weighting factor**
Sum	1
Product or Gain	2
Min/Max	2
Discrete Filter	5
Saturation	2
Fuzzy rule implication	4

Table 2. IE block weighting used for calculations

The *TE* index is related to the simplicity of the tuning procedures (as stated before, no adaptive or auto-tuning methods are considered in this work), while the *FTE* index considers the experience learned in this work while trying to obtain a marginal response improvement from an already stable condition.

An overall index has been designed to ponder all the previous indicators into a single value: *OPI* (Overall Pondered Index). The pondering proportions are arbitrary but still helpful to provide an idea at a glance. All the parameters are normalized prior to this calculation. The reason why the normalization is required is that the variation range for each of them is arbitrary, and may differ in several orders of magnitude between one and other (as it can be seen in the Comparative Analysis). This way, the normalized indexes are limited to the range [0, 1] and the pondering constants govern their contribution. The formulae used are:

$$OPI = 0.50 \cdot RI + 0.50 \cdot EI \tag{17}$$

$$RI = 0.35 \cdot NIAE + 0.40 \cdot NITAE + 0.25 \cdot NCE \tag{18}$$

$$EI = 0.25 \cdot NSF + 0.30 \cdot NIE + 0.30 \cdot NTE + 0.15 \cdot NFTE \tag{19}$$

Here, the N-prefix denotes the normalized index (e.g. *NIAE* for the normalized *IAE*) and is calculated as follows:

$$NP_i = \frac{P_i}{\sum_{k=1}^{4} P_k} \tag{20}$$

The subscript k = 1, 2, 3 and 4 refers to linear PID, Fuzzy PI case #1 (2 and 3 fuzzy sets defined over the input and output spaces, respectively), Fuzzy PI case #2 (7 fuzzy sets defined over the input and output spaces) and Hybrid Fuzzy PID, respectively.

4.2 Linear PID

A linear PID controller has been designed and implemented. The tuning has been carried out following the closed-loop Ziegler-Nichols method and the critical values obtained were *KU* = 0.5 and *TU* = 8.8 *s*. The PID gains and characteristic times have been determined to be *KP* = 0.3, T_I = 4.4 *s*, *KI* = 0.0682 s^{-1}, T_D = 1.1 *s* and *KD* = 0.33 *s*, as instructed by the method.

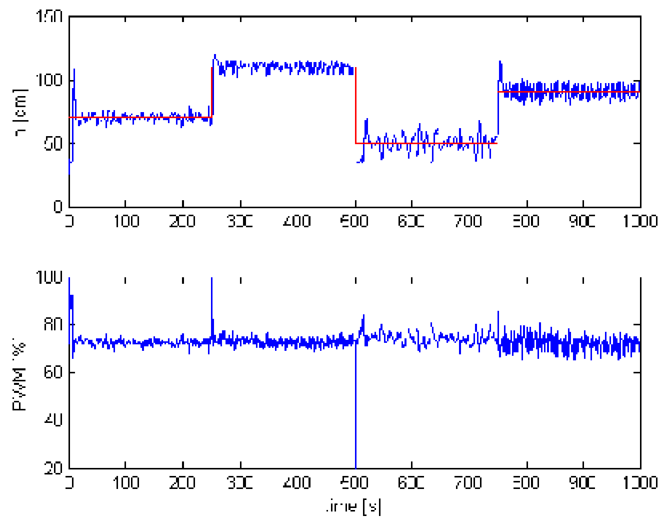

Fig. 2. Real-time results with the linear PID controller

4.3 Fuzzy PI

The linear PI controller parameters had to be calculated, since they are required for tuning purposes. Following the Ziegler-Nichols method, the resulting values obtained were *KP* = 0.225, T_I = 7.33 *s* and *KI* = 0.0307 s^{-1}.

After trial and error tests, the closed-loop system tuning was achieved for the case #2 controller. The final value of the controllers' parameters resulted in *KE* = 0.41, *KCE* = 2.9 and *KU* = 0.3.

These same parameters have also been applied to the case #1 controller, for comparison purposes.

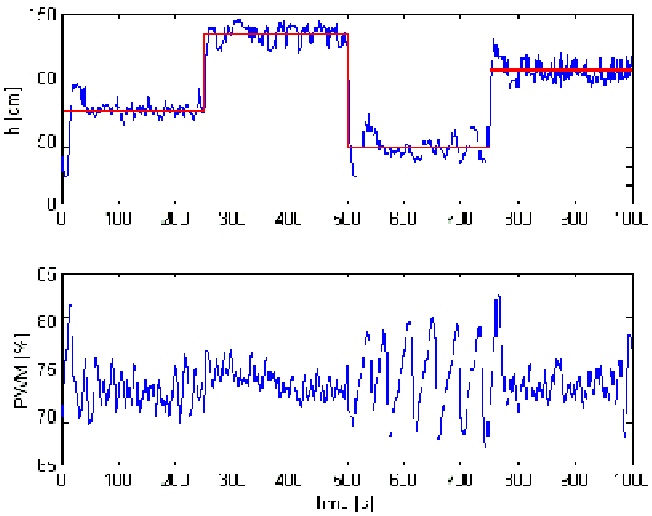

Fig. 3. Real-time results with the Fuzzy PI controller (case #2)

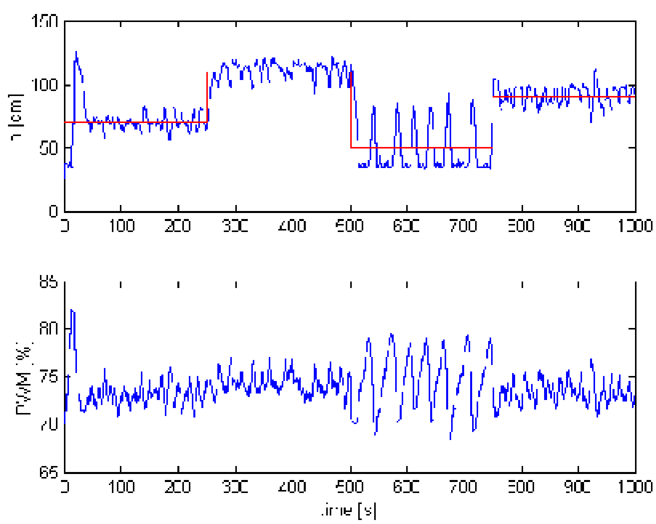

Fig. 4. Real-time results with the Fuzzy PI controller (case #1)

4.4 Hybrid Fuzzy PID

Using Equations (10) thru (13), the controllers' parameters were calculated and the following values were obtained: GE = 1, GU = 0.15, GCE = 2.2, GCU = 0.0136. Again, a fine-tuning was carried out and the structure was adjusted by modifying the consequence parameters to p_i = [0.9 1.1 1.0 0.8] and q_i = [1.0 1.2 1.0 0.8].

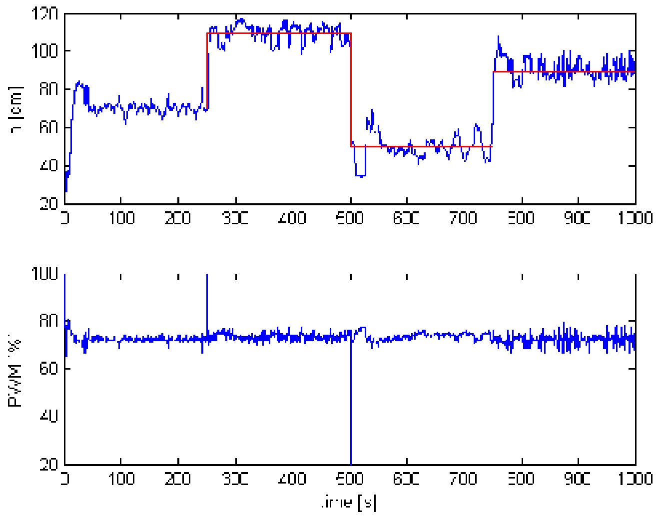

Fig. 5. Real-time results with the Hybrid Fuzzy PID controller

4.5 Comparative Analysis

The indexes explained earlier were calculated and the results are presented in Table 3 and Table 4.

Index	*Linear PID*	*Fuzzy PI (case #1)*	*Fuzzy PI (case #2)*	*Hybrid Fuzzy PID*
OPI	0.2407	0.2818	0.2679	0.2096
IAE	4.13E+03	9.63E+03	4.66E+03	4.47E+03
ITAE	2.15E+06	4.76E+06	2.09E+06	2.11E+06
CE	1.12E+04	9.47E+01	1.12E+02	1.17E+04

Table 3. Regulation indexes for the different controllers

The best *OPI* corresponds to the Hybrid PID controller. The Fuzzy PI controller (case #2) has exhibited the best steady state behavior (*ITAE*), being the linear PID the one that has shown the best *IAE* (transient response). The Fuzzy PI controller with 2 fuzzy sets (case #1) has revealed poor regulation abilities.

Index	*Linear PID*	*Fuzzy PI (case #1)*	*Fuzzy PI (case #2)*	*Hybrid Fuzzy PID*
SF	0.53	0.21	0.21	0.05
IE	19	124	472	87
TE	0.04	0.43	0.43	0.10
FTE	0.13	0.27	0.27	0.33

Table 4. Implementation indexes for the different controllers

On the other hand, related to the implementation indicators, the Fuzzy PI controllers show an increasing disadvantage. Also the tuning indicators for this kind of controllers are not very favorable.

5 Conclusions

According to the previous section, the best controller for this application has been the Hybrid Fuzzy PID, achieving differences in the general performance index between 15% and 35%. The superior and uniform behavior has played an important role, falling only 8% to 1% below the best figures in the corresponding categories.

Its flexibility and simplicity to be implemented and tuned have also contributed to this result. The implementation indexes show the Hybrid approach as the best fuzzy strategy. If compared to the linear controller, tuning and implementation effort is traded for structure flexibility.

The other fuzzy alternative requires a big number of fuzzy sets to produce comparable results. This significantly increases the implementation effort for no evident improvement. In addition to this, the tuning procedures for this kind of controllers are very informal if no adaptive techniques are used.

Finally, the linear PID has evidenced acceptable performance in both, the regulation and implementation areas. Nevertheless, certain non-linear plants might make this approach unsuitable.

Acknowledgments

The authors wish to thank the support of the Faculty of Engineering (University of Buenos Aires), CITEI (INTI) and the Faculty of Physics and Mathematics Sciences (University of Chile).

References

Cantrell T. (1994): **PID-Pong Challenge**. The Computer Applications Journal, #42, pp 60-65.

Cantrell T. (1994): **PID-Pong: Point, Set, Match: Using a Hitachi H8 for Real-time Control**. The Computer Applications Journal, #50, pp 68-73.

Cantrell T. (1996): **Fuzzy PID-Pong, The Final Chapter**. Circuit Cellar INK, #70, pp 68-80.
Cantrell T. (1996): **Fuzzy PID-Pong**. Circuit Cellar INK, #69, pp 90-97.
Chen G. (1995): **Conventional and Fuzzy PID Controllers: An Overview**. Department of Electrical and Computer Engineering, University of Houston, USA.
Choi B., Kwak S. and Kim B. (2000): **Design and Stability Analysis of Single-Input Fuzzy Logic Controller**. IEEE T-SMC 30 (2), pp 303-309.
Escamilla-Ambrosio P. and Mort N. (2002): **Auto tuning of Fuzzy PID controllers**. 15th IFAC, Spain.
Hu B., Mann G. and Gosine R. (1999): **New Methodology for Analytical and Optimal Design of Fuzzy PID Controllers**. IEEE T-FS 7 (5), pp 521-539.
Hu B., Mann G. and Gosine R. (2001): **A Systematic Study of Fuzzy PID Controllers – Function-Based Evaluation Approach**. IEEE T-FS 9 (5), pp 699-712.
Lee C. (1990): **Fuzzy Logic Controller – Part I, II**. IEEE T-SMC 20 (2), pp 404-435.
Lee C. and Teng C. (2001): **Tuning of PID Controllers for Stable and Unstable Processes with Specifications on Gain and Phase Margins**. International Journal of Fuzzy Systems, 3 (1), pp 346-355.
Li H. (1996): **Conventional Fuzzy Control and Its Enhancement**. IEEE T-SMC 26 (5), pp 791-797.
Li H. (1997): **A Comparative Design and Tuning for Conventional Fuzzy Control**. IEEE T-SMC 27 (5), pp 884-889.
Li H. and Gatland H. (1995): **A New Methodology for Designing a Fuzzy Logic Controller**. IEEE T-SMC 25 (3), pp 505-512.
Li H. and Tso S. (1999): **Higher Order Fuzzy Control Structure for Higher Order or Time-Delay Systems**. IEEE T-FS 7 (5), pp 540-552.
Li H., Gatland H. and Green A. (1997): **Fuzzy Variable Structure Control**. IEEE T-SMC 27 (2), pp 306-312.
Pedrycz W. (1995): **Fuzzy Control Engineering: Reality and Challenges**. 4th ICFS, Japan, pp 437-446.
Takagi T. and Sugeno M. (1985): **Fuzzy identification of systems and its applications to modeling and control**. IEEE T-SMC 15 (1), pp 28-44.
Tancara J. (1993): **Desarrollo y aplicación de controladores basados en lógica difusa**. Universidad de Tarapacá, Chile.
Tang K. (1987): **Comparing Fuzzy Logic with classical controller design**. IEEE T-SMC 17, pp 1085-1087.
Victor J. and Dourado A. (1997): **Adaptive Scaling Factors Algorithm for the Fuzzy Logic Controller**. Departamento de Engenharia Informático, Universidade Coimbra, Portugal.
Ying H., Siler W. and Buckley J. (1990): **Fuzzy Control Theory: A non-linear case**. Automatica 26 (3), pp 513-520.