A novel hybrid intelligent classifier to obtain the controller tuning parameters for temperature control

José Luis Calvo-Rolle[1], Emilio Corchado[2], Héctor Quintian-Pardo[2], Ramón Ferreiro García[1], Jesús Ángel Román[2] Pedro Antonio Hernández[3]

[1] Department de Ingeniería Industrial, Universidad de La Coruña
Avda. 19 de febrero, s/n, 15405, Ferrol, A Coruña, Spain
jlcalvo@cdf.udc.es, ferreiro@udc.es

[2] Departamento de Informática y Automática, Universidad de Salamanca
Plaza de la Merced s/n, 37008, Salamanca, Spain
escorchado@usal.es, hector.quintian@usal.es, zjarg@usal.es

[3] Departamento de Expresión Gráfica en la Ingeniería, Universidad de Salamanca
Av. Requejo, 33 Campus Viriato, 49022, Zamora, Spain
pedrohde@usal.es

Abstract. This study presents a novel hybrid classifier method to obtain the best parameters of a PID controller for desired specifications. The study presents a hybrid system based on the organization of existing rules and classifier models that select the optimal expressions to improve specifications. The model achieved chooses the best controller parameters among different closed loop tuning methods. The classifiers are based on ANN and SVM. The proposal was tested on the temperature control of a laboratory stove.

Keywords: Hybrid classifier, PID, closed-loop tuning, intelligent control.

1 Introduction

This study describes a hybrid classifier to obtain the controller parameters based on PID (Proportional-Integral-Derivative) closed-loop tuning. Although the PID controller is one of the most traditional types of controller, researchers are still working to improve its behaviour and performance [1-11]. There have been several studies with the same objective, but they have always been oriented to a specific system [5, 9, 11].

Nevertheless, there are many controllers operating well below the optimal state [14], overcoat controllers that are not self-tuning. It has therefore become critical to achieve new ways to solve this problem. Many studies related to the PID controller try to either establish optimal parameters according to the plant, or achieve self-tuning controller topologies [11, 12, 13].

The proposed topology described in this research has two phases. The first phase obtains characteristics of the plant response, while the second is applied to achieve the controller parameters by means of a hybrid classifier. The proposal makes it possible to achieve an intelligent topology with satisfactory results. The proposed topology only contemplates techniques with hard and previously tested implantation in the industry.

One of the aims of the implementation of the hybrid classifier is to contemplate the largest number of possibilities.

This study is organized as follows: section 2 provides a brief description of the general model; section 3 describes the tuning controller topology and briefly reviews the PID controller tuning in a closed-loop. Section 4 describes the hybrid classifier, section 5 presents empirical verification, and finally, section 6 provides conclusions and suggests future works.

2 Steps to obtain the best plant controller parameters

The procedure to obtain the best plant controller parameters to improve a given specification is illustrated in figure 1.

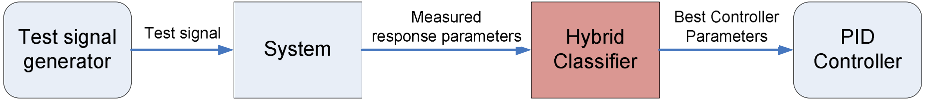


Fig. 1. Flowchart to obtain controller parameters.

As shown in figure 1, the first step in obtaining the best combination of the plant controller parameters involves a test signal generator. The test signal is then applied to the system. The next step is to measure the response characteristics. The best combination of the plant controller parameters is obtained by using the characteristics as input to the hybrid classifier, which supplies the controller with its parameters. In general terms, the hybrid classifier can be represented by the illustration in figure 2, where models based on rules [14-19] and machine learning techniques, such as Artificial Neural Networks (ANN) [24, 25] and Support Vector Machine [21, 22], are applied. After comparing the results, the best ones are chosen.

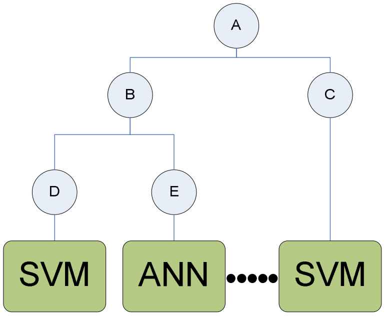


Fig. 2. Model final structure.

3 Tuning controller topology based on hybrid classifier

The topology proposed in this study is shown in figure 3. The following subsections describe the different aspects of the proposed topology.

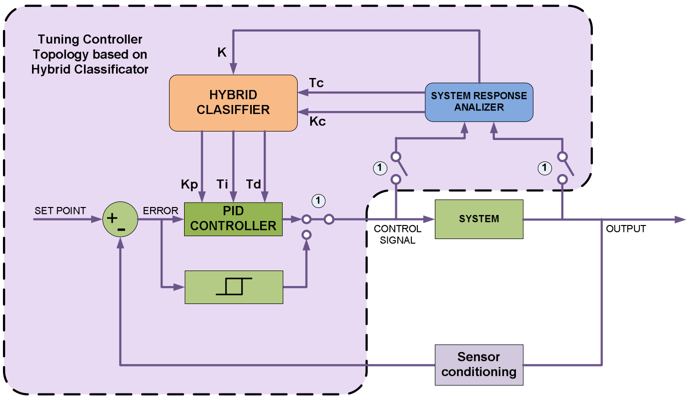


Fig. 3. Tuning controller topology.

3.1 PID controller format

There are several topologies for PID controllers, but in this study the standard format represented in equation 1 is used [14, 15].

$$u(t) = K\left[e(t) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\frac{de(t)}{dt}\right] \qquad (1)$$

where ‘*u*’ is the control variable and ‘*e*’ is the control error given by ‘*e* = *SP – y*’ (the difference between the set point ‘*SP*’ and conditioned output ‘*y*’). The other terms are the tuning controller parameters: proportional gain ‘*K*’, integral gain ‘*Ti*’ and derivate gain ‘*Td*’.

3.2 PID Controller Tuning in Closed-Loop

General procedure to calculate parameters. Two steps are necessary to obtain the controller tuning parameters in a closed loop:

- It is first necessary to set the system response to a permanent state of oscillation. Certain characteristics of the response must then be measured.
- According to the information gathered from the plant response, appropriate expressions must be applied to obtain correct controller parameters for the desired specifications.

Obtaining response characteristics in a closed-loop. Different methods can be used to obtain the controller parameter conditions. The present study uses the relay-feedback method proposed by Aström and Hägglud [14]. The results are very similar to those obtained by the traditional method proposed by Ziegler-Nichols [16]; however, the former offers some very important advantages, such as:

- The system operation is not nearly as unstable.
- The tuning process can be carried out at any time for any working point.

The implementation scheme of relay feedback is shown in figure 3 (switches ‘1’ to another position). A relay with hysteresis centred on a zero value with an amplitude *d* and a hysteresis window of width *h* is recommended for the general method.

The system oscillation has a period (*Tc*) with approximately the same as the Ziegler-Nichols method. The critical gain (*Kc*) of the process is obtained with equation 2, where *a* is the peak-to-peak value of the oscillation.

$$K_C = \frac{4d}{\pi\sqrt{a^2 - h^2}} \tag{2}$$

Obtaining controller parameters with formulas. After obtaining the *Tc* and *Kc* from the previous step, the controller parameters can be calculated. Many expressions have been developed by different authors [14, 16-19] with the aim, among others, of:

- Improving a particular specification of the system controlled response.
- Making the system robust to a particular criteria (Load Disturbance or Set Point Criteria)

There are several studies [14-19] that have developed different expressions. Even control equipment manufacturers have developed their own expressions according to their products line.

In this study, four methods (table 1) were taken into account: Ziegler-Nichols, Ziegler-Nichols some overshoot, Ziegler-Nichols without overshoot, and Tyreus-Luyben [16-19]. All of them are for Load Disturbance rejection criteria.

Table 1. Expressions of Controller parameters.

	Kp	Ti	Td	Appl. Range
Z-N	0.6 x Kc	0.5 x Tc	0.125 x Tc	2<k·Kc<20
Z-N Some Ov.	0.33 x Kc	Tc / 2	Tc / 3	2<k·Kc<20
Z-N Whitout Ov	0.2 x Kc	Tc	Tc / 3	2<k·Kc<20
Tyreus-Luyben	0.45 x Kc	2.2 x Tc	Tc / 6.3	2<k·Kc<20

4 Hybrid Classifier proposal

The proposed hybrid classifier is a fusion of rules and intelligent classification techniques. It can be divided into two different blocks:

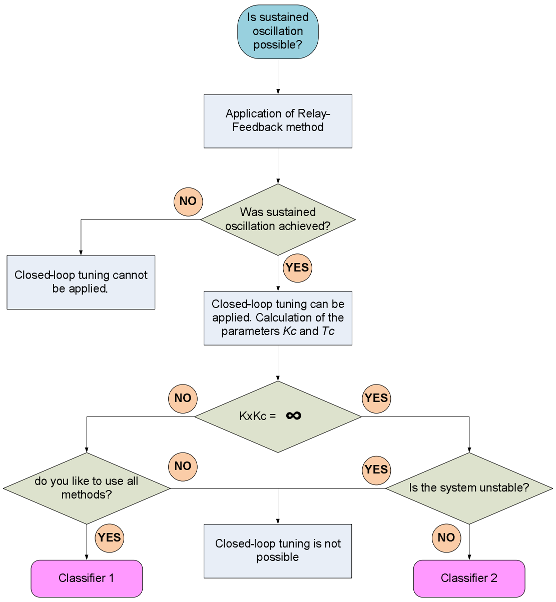
- Knowledge of existing rules (1st block). The aim of this block is to organize different rules for application ranges, authors expressions, criterions, and so forth.
- Intelligent classifiers (2nd block). This part of the model selects the most appropriate expressions to obtain the controller tuning parameters.

In general terms, this novel hybrid intelligent classifier selects the best tuning parameters, according to the system and the desired specifications of operation.

The next two subsections describe the hybrid model. The first shows the flowchart used to select the intelligent classifier. The second provides details of the classifiers.

4.1 Flowchart of knowledge of existing rules

After applying different methodologies of PID controller tuning in closed-loop, it is possible to obtain a flow-chart, as shown in figure 4. Many PID tuning rules in closed loop were taken into consideration to create this diagram, with the aim of achieving a generalized knowledge of the field. The following paragraphs explain the diagram in greater detail.



Is sustained oscillation possible?
Application of Relay-Feedback method
Was sustained oscillation achieved?
NO
YES
Closed-loop tuning cannot be applied.
Closed-loop tuning can be applied. Calculation of the parameters *Kc* and *Tc*
NO
KxKc = ∞
YES
do you like to use all methods?
NO
YES
Is the system unstable?
YES
Closed-loop tuning is not possible
NO
Classifier 1
Classifier 2

Fig. 4. Flowchart of knowledge of existing rules.

The flowchart is based on the premise that the plant engineer can set the system operation in sustained oscillation. As explained in section 3 there are several techniques to perform PID controller tuning in closed loop. The Relay-Feedback method is the most robust, making it possible to achieve better results. It is then necessary to determine if the system can achieve the sustained oscillation. If it is not possible, then this method cannot be applied to tuning in closed loop. Otherwise, with the system in this state it is necessary to calculate *Tc* and *Kc* parameters.

With *Tc* and *Kc* values it is possible to obtain the *K.Kc* indicator, where *K* is the gain of the process. This indicator, among other functions, defines the best expressions to achieve the desired system response.

The model must be able to know if the value of *K.Kc* is infinite or not. It is then possible to follow the flowchart in two ways:

- When *K.Kc* is not infinite, the operators must decide if they want to use all the expressions anyway. If this is not the case, the closed loop tuning is not applicable for the contemplated expressions; otherwise, it is possible to use the Classifier 1.
- When *K.Kc* is infinite, the operator must find out if the system is unstable. If it is unstable, closed loop tuning is not applicable. This means that the system has an integrator in its transfer function. It is then possible to use the Classifier 2.

4.2 Classifiers to complete the model

As shown in figure 4, with an organization of set rules, there are two blocks corresponding to the intelligent classifiers (1 and 2). Three techniques were applied to create these blocks: decision tree, artificial neural networks (ANN) and support vector machines (SVM). The following paragraphs describe the formulation of the model.

Model input. As seen in the flowchart of figure 4 and its description, *K,Kc* is a very important indicator. It defines, for instance, the application range of expressions. In all classifiers that were created, *K.Kc* was the input defining the system type, the system dynamics and, consequently, its controllability.

Dataset for model creation. As in other studies, it is necessary to select representative systems with the objective of generalizing the model as much as possible. Consequently, this study is based on [20]. This research includes a list of very representative systems, where all real systems behave in a similar fashion.

There is a problem with the initial dataset, which is that the *K.Kc* values of the initial systems have very close values; however the last systems have widely separated values. For this reason, many systems were created to solve the problem. To this end, a delay time is added to the 1th and 2nd order systems with Pade approximation [14]. Thus, a difference in the *K.Kc* values between consecutive systems is achieved, less than that of the unit. Finally 1704 systems were obtained to implement the classifiers. The systems obtained are balanced by controller parameter expressions.

Systems specifications for each expression. Each system is tested with the four expressions contemplated in table 1. Four specifications are then tested for system response to step input: response time (*Tr*), settling time (*Ts*), overshoot (*Ov*) and peak time (*Tp*). As a result, it is possible to obtain the expression that gives the best specification value. The present study used the following tuning methods: *Z&N* (Ziegler-Nichols), *Z&N wOv* (Ziegler-Nichols Whitout Overshoot), *Z&N sOv* (Ziegler-Nichols some Overshoot), *T&L* (Tyreus-Luyven).

Classification techniques analyzed to complete the model. Three techniques were taken into account to complete the model:

- Decision tree using the J48 learning algorithm: One of the classification methods contemplated in this research is the decision tree [23-25]. The decision tree approach is one of the most common approaches in automatic learning and decision making. The true purpose of decision trees is to classify the data into different groups, according to the dependent variable [23]. The decision trees were obtained by using the J48 algorithm [23, 26, 27]. The J48 algorithm was chosen because of its superior performance in most circumstances [27].
- Multilayer Perceptron (MLP): A multilayer perceptron is a feed forward artificial neural network [28]. It is one of the most typical ANNs due to its

robustness and relatively simple structure. However the ANN architecture must be well selected to obtain good results.

- Support Vector Machine (SVM): is a concept used in statistics and computer science for a set of related supervised learning methods that analyze data and recognize patterns. It is used for classification and regression analysis [21, 22] and trains a classifier by finding an optimal separating hyperplane which maximizes the margin between classes of data in a kernel-induced feature space [21, 22].

Classification results. For each specification (response time, settling time, overshoot and peak time) a classifier was created. Each set of expressions was categorized as follows:

- Class A: Z-N.
- Class B: Z-N Some Ov.
- Class C: Z-N Whitout Ov
- Class D: Tyreus-Luyben

Five different parameters were used to measure performance: Sensitivity (SE), Specificity (SPC), Positive Prediction Value (PPV), Negative Prediction Value (NPV) and Accuracy (ACC) (see equations from 3 to 7 respectively).

$$SE = \frac{TP}{(TP + FN)} \tag{3}$$

$$SPC = \frac{TN}{(FP + TN)} \tag{4}$$

$$PPV = \frac{TP}{(TP + FP)} \tag{5}$$

$$NPV = \frac{TN}{(TN + FN)} \tag{6}$$

$$ACC = \frac{TP + TN}{(TP + TN + FP + FN)} \tag{7}$$

where TP is the number of True Positive, TN is the number of True Negative, FN is the number of False Negative and FP is the number of False Positive.

In all cases k-fold cross-validation was used to split the dataset into a reasonable value that obtains good results. The k value is 10 for all models. For the decision tree classification the algorithm chosen was J48, which has the following configuration parameter values: 0.25 for the confidence threshold for pruning, and 2 for the minimum number of instances per leaf. In the case of MLP, tests were performed with 2 and 3 hidden layers, with the second value providing the best results. The number of neurons in hidden layers is within the range of 10-15. The activation functions tested in the hidden layer were: log sigmoid and tangent sigmoid. The tangent function achieved the best results. The activation function of the output layer is the log sigmoid. The

Winner Take All (WTA) technique was used to obtain the class provided by the MLP output. For SVM the selected kernel was the Gaussian radial basis function. For this technique, 15 different values were assigned for parameter γ (in a range from 2^{-12} to 2^{3}) and 17 different values for parameter C (from 2^{-5} to 2^{12}). A total of 255 (15x17) different combinations of parameters were taken into account.

Table 2. Percentage of correct classification for two Classifier Models

Model data	Training data	Response Time (Tr)			Overshoot (Ov)		
		J48	MLP	SVM	J48	MLP	SVM
Classifier 1	1704	81	93	**95**	83	92	**94**
Classifier 2	1704	78	94	**96**	86	**95**	93
Model data	Training data	Settling Time (Ts)			Peak Time (Tp)		
		J48	MLP	SVM	J48	MLP	SVM
Classifier 1	1704	87	**94**	91	79	91	**93**
Classifier 2	1704	83	**94**	93	80	93	**95**

Table 3. Confusion Matrix for classification rate of Ov of Classifier 2 using MLP.

	Method chosen by model			
Desired Method	A	B	C	D
A	**399**	9	15	3
B	9	**401**	6	10
C	10	18	**383**	15
D	17	12	19	**378**
TP	399	401	383	378
TN	1242	1239	1238	1250
FP	36	39	40	28
FN	27	25	43	48
SE	0,937	0,941	0,899	0,887
SPC	0,972	0,969	0,969	0,978
PPV	0,917	0,911	0,905	0,931
NPV	0,979	0,980	0,966	0,963
ACC	0,963	0,962	0,951	0,955

Table 2 shows the percentage of correct classification using the previously mentioned techniques for the two classifiers. In each case, the selected classifier is the one that achieves the best percentage of correct classification (table 2 values in bold).

For all the cases considered in table 2, the best configuration for each technique used was selected. The confusion matrix was created in each case. An example of confusion matrix is shown in table 3 where Overshoot (Ov) is tested for Multi Layer Perceptron case.

5 Empirical verification with a physical plant

An empirical verification of the Hybrid Classifier presented in this study was performed at a laboratory plant (figure 5) in which the temperature is controlled by adjusting the power provided to the heater element inside.

5.1 The physical description

The temperature variable depends on the following parameters: T1(t) is the temperature measured outside the stove; V is the air volume in the stove; SP(t) is the set point for the desired temperature; T2(t) is the measured temperature in the recipient; u(t) is the signal control to operate the heating element; Kv and Kt are constants related to the features of the heating element properties and the temperature sensor respectively.

5.2 Implementation of the control

The test was performed in the Labview® environment. For operations at the plant, a National Instruments data acquisition card (model USB-6008 12-bit 10 KS /s Multifunction I/O) was chosen. The diagram of the process is implemented in Labview® editor with the control Scheme shown in figure 6. Different gain blocks were added to adapt signals to all operation range.

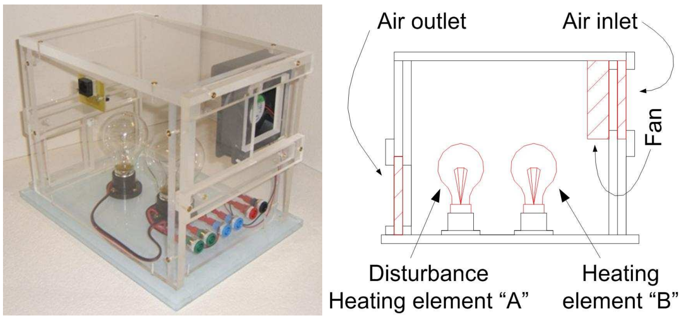


Fig. 5. Photograph and scheme of the real plant

It was necessary to add a filter block with an edge frequency of 1.5 rad/sec (9.5Hz) in order to reduce noise from the analog input. Using a switch it is possible to select either a PID control or Relay-Feedback configuration. Figure 7 shows the internal implementation of a PID block. PID controller gains (Ti, Td, and Kp) are programmed manually.

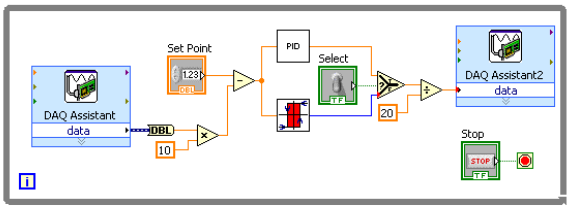


Fig. 6. Control scheme implemented in Simulink

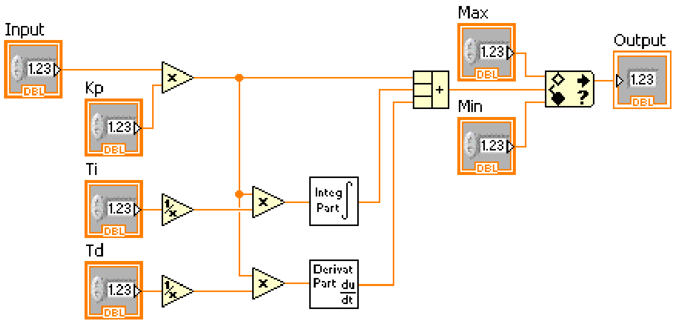


Fig. 7. PID block internal scheme

5.3 Practical behavior of Hybrid Classifier application onto the plant

Some tests were performed on the physical system previously described in section 5.1, with the aim of checking the behavior of the novel Hybrid classifier model.

If the hybrid classifier is applied to the physical system with a set point of 35 ºC and the temperature outside is 20 ºC, the expressions chosen are Tyreus-Luyben. The system response with a step input for the working point of the design is shown in figure 8. Figure 9 shows the response with a step input for a traditional Ziegler-Nichols method in closed loop. As seen in these figures, the hybrid classifier model takes the best expressions contemplated in this study to achieve the best overshoot. Perturbations are introduced in both cases, first with the fan and then with the heating element "A", both of which are included in the stove. Comparing figures 8 and 9, it is apparent that the first method (from the hybrid classifier) is more robust.

6 Conclusions

A Hybrid Classifier was presented in this study, and tested with a practical application where a PID controller was used. The best advantage offered by the method is that it ensures the most appropriate selection of the contemplated expressions in order to obtain the PID controller parameters.

It should be noted that the novel model is easy to expand to other system types, such as: level control, pressure control, humidity control, and so forth. At that point, it is only necessary to contemplate the new systems in order to create the Hybrid Classifiers. The rest of the model is completely valid.

With the aim of obtaining the best results, more than one technique was taken into account. The final classifier, among the typical rules of PID controller, includes two techniques in the final stage (ANN and SVM), according to where they achieve the best results.

Several future research lines will be considered, two of which are of particular importance. Firstly it is necessary to consider the responses of the physical plant in creating the hybrid classifier. The other important future research line is to perform tests with real industrial plants and include them in the adaptive mechanism model in order to create a more versatile the tool.

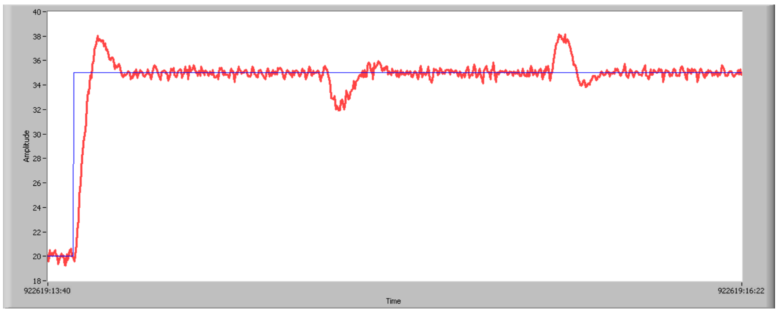


Fig. 8. System response with method from Hybrid Classifier

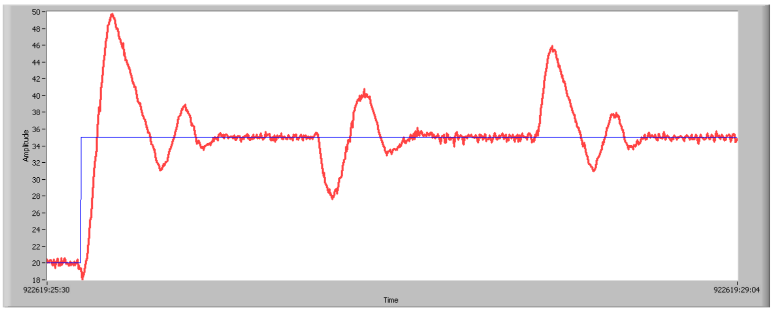


Fig. 9. System response with traditional Ziegler-Nichols method in closed-loop

Acknowledgments. This research is partially supported by projects TIN2010-21272-C02-01 from the Spanish Ministry of Science and Innovation. The authors would also like to thank the manufacturer of components for vehicle interiors, Grupo Antolin Ingeniería, S.A. which provided support through MAGNO 2008 – 1028 – CENIT funded by the Spanish Ministry of Science and Innovation.

References

1. Hsu C., Chen G., Lee T.: Robust intelligent tracking control with PID-type learning algorithm. Neurocomputing, 71, 234-243 (2007)
2. Gottwald S.: Mathematical Fuzzy Control. A Survey of Some Recent Results. Logic Journal of IGPL, 13, 525-541 (2005)
3. Zhang J., Zhuang J., Du H., Wang S.: Self-organizing genetic algorithm based tuning of PID controllers. Information Sciences, 179, 1007-1018 (2009)
4. Juang Y., Chang Y., Huang C.: Design of fuzzy PID controllers using modified triangular membership functions. Information Sciences, 178, 1325-1333 (2008)
5. Liu H., Coghill G.: A model-based approach to robot fault diagnosis. Knowledge-Based Systems, 18, 225-233 (2005)
6. Lu H., Chang J., Yeh M.: Design and analysis of direct-action CMAC PID controller. Neurocomputing, 70, 2615-2625 (2007)
7. Sala A., Cuenca Á., Salt J.: A retunable PID multi-rate controller for a networked control system. Information Sciences, 179, 2390-2402 (2009)
8. Sumar R.R., Coelho A.A.R., Coelho L.D.S.: Computational intelligence approach to PID controller design using the universal model. Information Sciences, 180, 3980-3991 (2010)

9. Thangaraj R., Chelliah T.R., Pant M., Abraham A., Grosan C.: Optimal gain tuning of PI speed controller in induction motor drives using particle swarm optimization. Logic Journal of IGPL, 19, 343-356 (2010)
10. Juang Y., Chang Y., Huang C.: Design of fuzzy PID controllers using modified triangular membership functions. Information Sciences, 178, 1325-1333 (2008)
11. Ye J.: Adaptive control of nonlinear PID-based analog neural networks for a nonholonomic mobile robot. Neurocomputing, 71, 1561-1565 (2008)
12. Romero J.A., Sanchis R., Balaguer P.: PI and PID auto-tuning procedure based on simplified single parameter optimization. Journal of Process Control, 21, 840-851 (2011)
13. Sun J., Zhang D., Li X., Zhang J., Du D.: Smith Prediction Monitor AGC System Based on Fuzzy Self-Tuning PID Control. International Journal of Iron and Steel Research, 17, 22-26 (2010)
14. Astrom K.J., Hagglund T.: Advanced PID Control. Pearson Education. Madrid. (2009)
15. Feng Y.L., Tan K.C.: Process identification and PID Control. John Wiley&Sons. USA. (2009)
16. Visioli A.: Practical PID Control. Springer-Verlag, London. (2010)
17. Johnson M.A., Moradi M.H.: PID Control: New identification and Design methods. Springer-Verlag, London. (2010)
18. Seki H., Shigemasa T.: Retuning oscillatory PID control loops based on plant operation data. J. Process Control, 20, 217–227 (2010)
19. Tyreus B.D., Luyben W.L.: Tuning PI controllers for integrator/dead time processes. Industrial Engineering Chemistry Research, 11, 2625-2628 (1992)
20. Astrom K.J., Hagglund T.: Benchmark Systems for PID Control. Preprints FAC Workshop on Digital Control. Past, present and future of PID Control. 181-182, Tarrasa (2000)
21. Vapnic V.: The nature of statistical learning theory. Springer-Verlag. New York (1995)
22. Cristianini N., Shawe-Taylor J.: An Introduction to Support Vector Machines. Cambridge University Press, Cambridge (2000)
23. Parr O.: Data Mining Cookbook. Modeling Data for Marketing, Risk, and Customer Relationship Management. John Wiley & Sons, Inc. New York. (2001)
24. Duda R.O., Hart P.E., Stork D.G.: Pattern Classification. John Wiley & Sons, Inc. Canada (2001)
25. Mitchell T. M.: Machine Learning. McGraw-hill, New York (1997)
26. Frank E., Witten I.: Data Mining: Practical Machine Learning Tools and Techniques. Second Edition. Morgan Kaufmann. (2005)
27. Rokach L., Maimon O.: Data Mining with Decision Trees: Theory and Applications. World Scientific Publishing. USA. (2008)
28. Alpaydin E.: Introduction to Machine Learning. The MIT Press. Oxford, (2009)