QUALITY PREDICTION IN INDUSTRIAL PROCESSES: APPLICATION OF A NEURO-FUZZY SYSTEM

Rui Pedro Paiva[a], **António Dourado**[a], **Belmiro Duarte**[b]

[a]*CISUC – Centro de Informática e Sistemas da Universidade de Coimbra*
Departamento de Engenharia Informática, PÓLO II da Universidade de Coimbra, Pinhal de Marrocos, P 3030, Coimbra, Portugal
Tel. +351-239-790000, Fax: +351-239-701266,
ruipedro@dei.uc.pt, dourado@dei.uc.pt

[b]*Companhia de Celulose do Caima, S A,*
P 2250 Constância, Portugal

Abstract: In chemical industries, as paper pulp, quality control is a decisive task for competitiveness. Quality prediction is determinant in quality control. However the complexity of the production processes, their non-linear and time varying characteristics does not allow to develop reliable prediction models based on first principles. New tools issued from fuzzy systems and neural networks are being developed to overcome these difficulties. In this paper a neuro-fuzzy strategy is proposed to predict bleaching quality by predicting the outlet brightness. Firstly, a fuzzy subtractive clustering technique is applied to extract a set of fuzzy rules; secondly, the centers and widths of the membership functions are tuned by means of a fuzzy neural network trained with backpropagation. This technique seems promising since it permits good results with large nonlinear plants. Furthermore, it describes the plant using a set of linguistic rules which have the advantage of being closer to natural human language, so, more intuitive for operators.

Keywords: pulp industry, fuzzy modelling, neural-networks modeling.

1. INTRODUCTION

Pulp and paper industry is actually subjected to a very high demand concerning quality. Pulp bleaching, one of its sub-processes, is a nonlinear process for which there are still poorly understood phenomena. It is a sequential process influenced by a large number of variables for which the individual influences are not well known. First principles models do not exist with a sufficient level of accuracy. This leads to the development of other kind of approaches, such as fuzzy systems and neural networks. Fuzzy systems are general approximators, capable of accurately representing nonlinear processes and including a priori existent knowledge about the process (Pedrycz and Waletzky, 1997). Fuzzy systems can also be developed from plant data, but in this case techniques for establishing the fuzzy rules and tuning them are needed. One way to deal with these needs is through fuzzy clustering (Delgado *,et al.*, 1998). However one must face the problem with the *curse of dimensionality* of the rule base. This problem has been faced by several authors (see for example Wang and Rong (1999), Runkler (1998)).

One important characteristics on the bleaching process is its time-varying transport delay, that is faced in this work.

The strategy is developed in two phases: firstly, subtractive clustering is applied to extract a set of fuzzy rules; secondly the membership functions (centers and widths) are learned by a neural networks trained by backpropagation. The obtained results are promising, showing the potential of the technique. Furthermore, it describes the plant by a set of interpretable linguistic (fuzzy) rules close to the natural human language and intuitive for the operators.

This paper is organized as follows. Section 2 describes the two-phase algorithm for fuzzy prediction. Section 3 describes briefly of the pulp bleaching plant. Section 4 is devoted to the description of a possible strategy for dealing with the variable time delays. Finally, Section 6 concludes the paper pointing out the advantages and limitations of the strategy used and the main problems encountered, as well as some directions for future work, and a possible strategy to include the variable time delays in the model.

2. THE FARX PREDICTOR STRUCTURE AND SUBTRACTIVE CLUSTERING

Given a set of operating data and, possibly, an initial set of linguistic rules dictated by experts, a Fuzzy Auto-Regressive with Exogenous (FARX) variable structure is followed to model the bleaching plant. The system is described by a set of rules of Mamdani inference type (1):

$$\begin{aligned} R_i: If \quad & y_1(k) \text{ is } A_{11\,i} \text{ and } \ldots \text{and } y_q(k-n_q) \text{ is } A_{qn_q i} \\ \text{and } & \ldots u_1(k-d_1) \text{ is } B_{11\,i} \text{ and } \ldots \text{and } u_p(k-m_p-d_p+1) \text{ is } B_{pm_p i} \\ \text{then } & y_1(k+1) \text{ is } C_{1i} \text{ and } \ldots \text{and } y_q(k+1) \text{ is } C_{qi} \end{aligned} \quad (1)$$

where q is for number of system outputs, p the number of inputs and the parameters n_1, ..., n_q, m_1, ..., m_p, d_1, ..., d_p are related to the system order and discrete pure time delay. A_{jki} , B_{jki} and C_{ji} are the fuzzy sets (linguistic values) for each output and input variables, defined by their membership functions: $\mu_{A_{jki}}, \mu_{B_{jki}}, \mu_{C_{ji}}$, $i = 1, 2, ..., R$.

The parameters n_1, ..., n_q, m_1, ..., m_p, d_1, ..., d_p are properly chosen on the basis of prior knowledge or by comparison of different values in terms of some criteria. Assuming this problem is solved, the issue is: (i) to obtain a set of rules of type (1); (ii) to adjust the parameters of the membership functions using data collected from the system (2):

$$X=[\theta(1) \ \cdots \ \theta(N-1)]^T, \ \Psi=[Y(1) \ \cdots \ Y(N-1)]^T \quad (2)$$

where N is the number of data samples available for the identification purpose and q is the regression vector.

2.1. Subtractive Clustering

In order to obtain a set of R rules avoiding the problems inherent to grid partitioning, e. g., rule base explosion, subtractive clustering is applied (Chiu, 1994). This technique is employed since it allows a scatter input-output space partitioning.

Subtractive clustering is, essentially, a modified form of the Mountain Method. Thus, let Z be the data set obtained by concatenation of the sets X and Y (2). Assuming that all the data points are normalized in each dimension, the data set Z is bounded by a hypercube. In the algorithm, each point is seen as a potential cluster center, for which some measure of potential is assigned (3):

$$P_i = \sum_{j=1}^{N} e^{-\alpha \left\| z_i - z_j \right\|^2} \quad (3)$$

where $\alpha=4/r_a^2$ and $r_a>0$ defines the neighborhood radius for each cluster center. Therefore, the potential associated to each cluster depends on its distance to all of the points, leading to clusters with high potential where the neighborhoods are dense.

After computing the potential for each point, the one with higher potential is selected as the first cluster center. Let z_1^* be the center of the first group and P_1^* its potential. Then, the potential for each point z_i^* is reduced, especially for the points closer to the center of the cluster (4):

$$P_i \Leftarrow P_i - P_1^* e^{-b \left\| z_i - z_1^* \right\|^2} \quad (4)$$

where $b=4/r_b^2$ and $r_b>0$ represents the radius of the neighborhood for which significant potential reduction will occur. The radius for reduction of potential should be to some extent higher than the neighborhood radius to avoid closely spaced clusters. Typically, $r_b=1.5r_a$. Since the points closer to the cluster center will have their potential strongly reduced, the probability for those points to be chosen as the next cluster is lower. This procedure (selecting centers and reducing potential) is carried out iteratively, until the stopping criteria is reached:

If $P_k^* > e^{up} P_1^*$
  Accept z_k^* as the next cluster center
    and continue
Otherwise,
  If $P_k^* < e^{down} P_1^*$
    Reject z_k^* and finish the algorithm.
  Otherwise
    Let d_{min} be the shortest distance between z_k^* and all the centers already found
    If $d_{min}/r_a + P_k^*/P_1^* \geq 1$

Accept z_k^* as the next cluster center and continue
Otherwise
Reject z_k^* and assign it the **potential 0.0**.
Select the point with **higher potential** as new z_k^*.
Repeat the test.
End If
End If
End If

There, e^{up} specifies a threshold above which the point is selected as a center, without any doubts and e^{up} specifies the threshold below which the point is definitely rejected. The third case is where the point is characterized by a good trade-off between having a sufficiently high potential and being distant enough from the clusters determined before. Typically, $e^{up}=0.5$ and $e^{up}=0.15$.

By the end of clustering, a set of fuzzy rules will have been obtained. Each cluster will represent a rule. However, since the clustering procedure is conducted in a multidimensional space, fuzzy sets must be obtained. As each axis of the multidimensional space refers to a variable, the centers of the membership functions for that variable are obtained by projecting the center of each cluster in the corresponding axis. As for the widths, they are obtained on the basis of the neighborhood radius, r_a, defined while performing subtractive clustering. Since Gaussian membership functions are used, their standard deviations are computed by (5):

$$\sigma_{ij} = r_a \cdot \frac{\max(z_{kj}) - \min(z_{kj})}{\sqrt{8}}, k = 1,...,N \quad (5)$$

2.2. Self-Organization by Fuzzy Neural Network

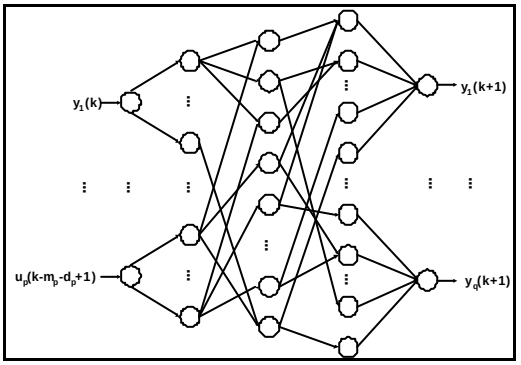

Figure 1: Structure of the fuzzy neural net.

After deriving an initial fuzzy inference system based on fuzzy clustering, its parameters, i.e., the centers and widths of membership functions must be optimized. In this paper, this is accomplished by means of training a fuzzy neural network (FNN) using standard backpropagation.

The structure of the FNN is presented in Figure 1. This structure can be found in (Lin, 1995). There, Gaussian membership functions are used. In the present work, two-sided membership functions are used, in order to allow more flexibility. The fuzzy neural network consists of five layers, which are described as follows.

Layer 1 contains the input nodes, which represent input linguistic variables. This layer simply passes the inputs to layer 2.

The nodes in layer 2 are the linguistic terms of each input variable, represented by Gaussian membership functions. This layer is responsible for the fuzzification of the crisp input values (6):

$$a_j^{(2)} = \begin{cases} e^{-\frac{(x_i - c_{ijL})^2}{s_{ijL}^2}}, & x_i < c_{ijL} \\ 1, & c_{ijL} \le x_i \le c_{ijR} \\ e^{-\frac{(x_i - c_{ijR})^2}{s_{ijR}^2}}, & x_i < c_{ijR} \end{cases} \quad (6)$$

where $a_j^{(2)}$ denotes the activation for each node on the second layer, c_{ijL} and c_{ijR} stand for the left and right centers of a two-sided Gaussian, s_{ijL} and s_{ijR} refer to the left and right standard deviations and x_i represents the i-th input. In the following, the superscript will always stand for the layer number.

In layer 3, each node is assigned to a rule of the fuzzy inference system. The antecedents of each rule are defined by setting proper links form nodes at layer 2 to nodes at layer 3. This layer fires each rule based on some fuzzy AND operation. In this work, the truncation operator *min* was used. Normally, an algebraic operator, like *product*, should be used in order to apply the gradient for training the neural network. The two approaches were tested and, since better results were obtained with the operator *min*, despite not being continuous, the operator referred was selected. The output of the third layer is as follows (7):

$$a_k^{(3)} = min\left(a_{j_1}^{(2)}, a_{j_2}^{(2)}, ..., a_{j_n}^{(2)}\right) \quad (7)$$

Since there are some rules that share the same consequent, layer 4 integrates those rules using some fuzzy OR operation. The nodes at layer 4 define the linguistic terms for each output, represented by Gaussian membership functions, as in layer 2. For the same reason as in layer 3, a truncation operator, namely *max*, was used (8):

$$a_l^{(4)} = max\left(a_{k_1}^{(3)}, a_{k_2}^{(3)}, ... a_{k_n}^{(3)}\right) \quad (8)$$

Layer 5 is the output layer. The role of this layer is to perform defuzzification, i.e., convert fuzzy numbers into crisp numbers. In this work, an adaptation of the

center of area defuzzification method is used, in order to cope with two-sided Gaussian functions and to incorporate the effect of the widths into the defuzzification strategy (9):

$$a_m^{(5)} = \frac{\sum_l \frac{1}{2}\left(c_{lmL}\, s_{lmL} + c_{lmR}\, s_{lmR}\right) a_l^{(4)}}{\sum_l \frac{1}{2}\left(s_{lmL} + s_{lmR}\right) a_l^{(4)}} \tag{9}$$

As in layer 2, c_{lmL}, s_{lmL}, c_{lmR}, and s_{lmR} represent the left and right parameters of the two-sided Gaussian membership function.

As stated before, the objective of the presented FNN is to perform optimization of the centers and widths of the Gaussian membership functions. For that matter, supervised learning is carried out based on acquired data (2), using standard backpropagation. The goal is to minimize the error function (10):

$$E_m = \frac{1}{2}\left[y_m - a_m^{(5)}\right]^2 \tag{10}$$

where y_m stands for the desired network output and $a_m^{(5)}$ is the actual network output for the m-th output. Assuming that w_{ij} is the parameter to adjust, the general learning rule is as (11):

$$\Delta w_{ij} = lr \cdot \left(-\frac{\partial E_m}{\partial w_{ij}}\right) \tag{11}$$

where *lr* is the learning rate.

Based on equations (10) and (11), the expressions for adapting the centers and widths of the membership functions are presented below.

Layer 5. In this layer, the centers and widths of the output membership functions are updated. This is conducted by (12), (13) and (14).

$$d_m^{(5)} = y_m - a_m^{(5)} \tag{12}$$

$$\frac{\partial E_m}{\partial c_{lmL}} = -d_m^{(5)} \cdot \frac{s_{lmL}\, a_l^{(4)}}{\sum_r \left(s_{rmL} + s_{rmR}\right) a_r^{(4)}} \tag{13}$$

$$\frac{\partial E_m^{(\,)}}{\partial s_{lmL}} = -d_m^{(5)} \cdot \frac{c_{lmL} a_l^{(4)} \sum_r \left(s_{rmL} + s_{rmR}\right) - a_l^{(4)} \sum_r \left(c_{rmL} s_{rmL} + c_{rmR} s_{rmR}\right)}{\left[\sum_r \left(s_{rmL} + s_{rmR}\right)\right]^2} \tag{14}$$

Layer 4. In this layer, there are no parameters to update. Therefore, only the error signals (δ) need to be computed for backpropagation (15):

$$d_l^{(4)} = \sum_m d_m^{(5)} \frac{\left(c_{lmL} s_{lmL} + c_{lmR} s_{lmR}\right) \sum_r \left(s_{rmL} + s_{rmR}\right) - \left(s_{lmL} + s_{lmR}\right) a_l^{(4)} \sum_r \left(c_{rmL} s_{rmL} + c_{rmR} s_{rmR}\right)}{\left[\sum_r \left(s_{rmL} + s_{rmR}\right)\right]^2} \tag{15}$$

Layer 3. As in layer 4 only the error signals need to be computed (16):

$$d_k^{(3)} = \sum_l d_l^{(4)} \frac{\partial a_l^{(4)}}{\partial a_k^{(3)}} \tag{16}$$

Layer 2. In layer 2, the centers and widths of the input membership functions are updated according to eqs. (17), (18), (19), (20) and (21).

$$\frac{\partial E_m^{(\,)}}{\partial c_{ij}} = \left(\sum_k d_k^{(3)} \frac{\partial a_k^{(3)}}{\partial a_j^{(2)}}\right) \frac{\partial a_j^{(2)}}{\partial c_{ij}} \tag{17}$$

$$\frac{\partial E_m^{(\,)}}{\partial s_{ij}} = \left(\sum_k d_k^{(3)} \frac{\partial a_k^{(3)}}{\partial a_j^{(2)}}\right) \frac{\partial a_j^{(2)}}{\partial s_{ij}} \tag{18}$$

$$\frac{\partial a_j^{(2)}}{\partial c_{ijL}} = \frac{2\left(x_i - c_{ijL}\right)}{s_{ijL}^2} e^{-\frac{\left(x_i - c_{ijL}\right)^2}{s_{ijL}^2}} \tag{19}$$

$$\frac{\partial a_j^{(2)}}{\partial s_{ijL}} = \frac{2\left(x_i - c_{ijL}\right)^2}{s_{ijL}^3} e^{-\frac{\left(x_i - c_{ijL}\right)^2}{s_{ijL}^2}} \tag{20}$$

$$\frac{\partial a_k^{(p3)}}{\partial a_j^{(p2)}} = \begin{cases} 1 & , a_k^{(p3)} = a_j^{(p2)} \\ 0 & , a_k^{(p3)} \neq a_j^{(p2)} \end{cases} \tag{21}$$

3. PULP BLEACHING PLANT

Bleaching is done to decolorize the lignin present in wood fibbers. Chemicals are added, which react with the unbleached chromophores producing the desired bleached chromophores so that pulp properties can satisfy the standards demanded by paper industry. A major concern is to obtain satisfactory outlet brightness.

The present plant uses a Totally-Chlorine Free (TCF) bleaching sequence. Some TCF sequences have been used in the past years. In our case an EOP/P sequence is conducted, as presented in Figure 2 (Caima, 1994).

3.1. The Process

After cooking the wood with acid for delignification, washing and screening the pulp, the bleaching stage is ready to begin. First of all, the pulp is washed in washers 1 and 2. Then, in the EOP (**E**xtraction with NAOH, **O**xygen and Hydrogen **P**eroxide) stage the pulp is mixed with chemicals, namely hydrogen peroxide, oxygen (bleaching agents), caustic soda (to adjust the pH of the reacting mixture), and sodium silicate (peroxide stabilizer). This mixture reacts within towers 1 and 2 for approximately 4 hours. Before the P (extraction with Hydrogen **P**eroxide) stage, the pulp is washed in washer 3 in order to recover chemicals and energy. In the P stage the same chemicals as before, except for oxygen are added.

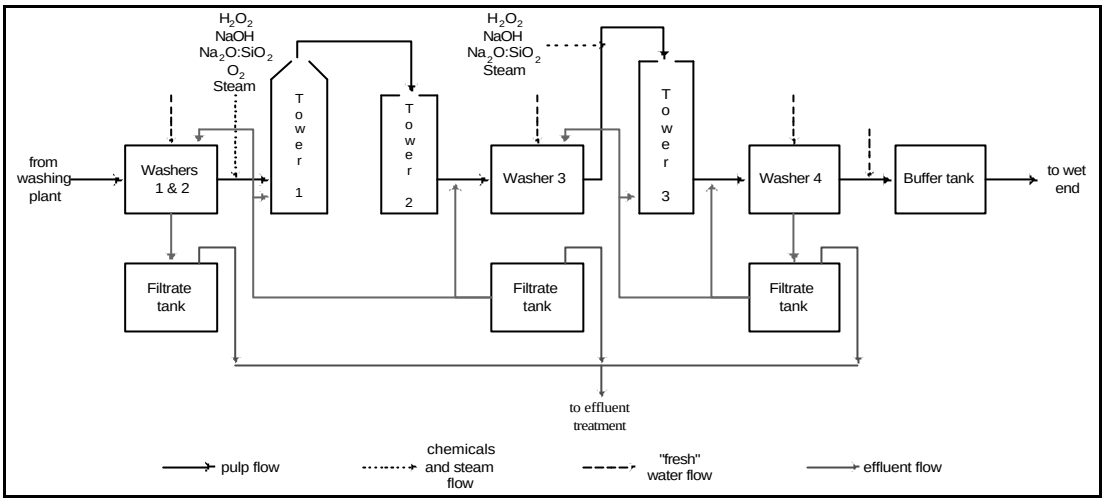

Figure 2: The EOP/P sequence.

The reaction takes place in tower 3 for approximately 2 hours. After this residence time, the pulp is washed in washer 4 and is then conducted to the drying section where it stays for about 1 hour. The total bleaching time, from washers 1 and 2 until dried pulp is obtained takes about 8 hours.

The final bleaching quality is influenced by a great deal of variables. According to experts' knowledge the variables that have a stronger influence on the final pulp quality are inlet brightness, inlet pulp flow, inlet permanganate number (which is a lignin concentration measurement), hydrogen peroxide in both of the stages and inlet pulp flow.

3.2. Brightness Analysis for Quality Classification

There are a few high-level rules that give some insight on the final brightness achieved: it increases with peroxide flow; it increases with pH; it increases with the consistency; it increases with temperature, until some threshold; it increases with inlet pulp; it decreases with inlet permanganate number. This information can be compared with the set of linguistic rules obtained by the fuzzy inference system.

In (Duarte, 1995), information on the delay times relating each input variable and the outlet brightness is presented. There, it is said that the delay time from inlet brightness to outlet brightness is 7-8 hours, which corresponds to the bleaching time referred above. For inlet pulp flow and inlet permanganate number the delay time should be the same. Concerning the peroxide flow in the P stage, the effect of a change on it affects outlet brightness from 3 to 5 hours later. For the peroxide flow in the EOP stage, the delay time should correspond to time elapsed since inlet pulp is washed in washers 1 and 2. So, a delay time of 6.5-7.5 hours is assumed.

4. RESULTS

Some of the measured variables are not sufficiently excited. Thus, their contribution for the achieved bleaching quality is not easily assessed only with measurements. Moreover, according to the experts' experience, the most important input variables are peroxide flow, inlet brightness and pH. Therefore, these are the input variables used to model the plant. Some experiments were carried out with the full set of variables. However, the inclusion of those variables did not bring any better results (actually, some cases happened to worsen the model).

The fuzzy inference system is obtained from the input-output measurements using subtractive clustering and tuning the membership functions with the algorithm in section 2. The sampling interval was defined in the mill as one hour; this sampling interval seems to be sufficient since the system's dynamics are very slow. Simulations were carried out with N=976 training samples. The parameter r_a for subtractive clustering was defined with the value r_a =0.35, leading to R=53 rules. Figure 3 presents the training results and Figure 4 shows model validation. There, the continuous line represents real process data, whereas the dashed line represents the model output. The variable depicted is the final brightness achieved, as described in section 3.1.

For the training data, the root mean square error (rmse) was equal to 0.165. However, for the validation data the rms error is slightly higher: rmse = 0.254. We can, therefore, conclude that the model obtained does not have satisfactory generalization capabilities. Some possible reasons for that are noise in measurements, inadequate sampling intervals or inconsistent training and validation sets, resulting from the variable time delay of the system. As stated above, the total pulp residence time varies from 7 to 10 hours (depending on the inlet pulp flow), according to the experts. The described technique

seems not to be able to satisfactorily cope with this situation. Thus, a strategy for capturing the effect of the variable time delay is needed. The inlet pulp flow and the levels in the towers mainly influence this delay. Consequently, it is the authors' opinion that including those variables in the model would make it possible to capture the transport delays. Including those variables and extra input variable regression would give the neuro-fuzzy system enough information to find a proper structure, i.e., select the right past input from the regression set, based on measurements for the levels and inlet pulp. However, as can be seen, this scheme was not completely successful, perhaps due to the absence of measurements for the level in tower 1. Yet, if one has present the difficulties and uncertainties related to the pulp bleaching plant and the (strong) presence of noise in the industrial environment, an rms error of 0.254, can be found satisfactory.

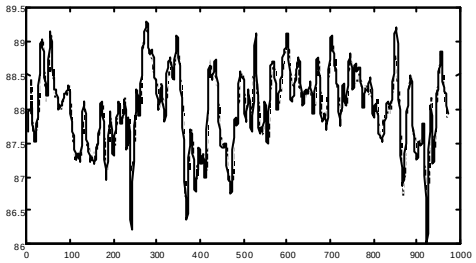

Figure 3. Validation : —— Process - - - FNN output

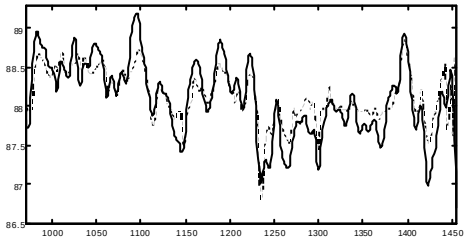

Figure 4. Validation : —— Process - - - FNN output

5. CONCLUSIONS

A predictor for the output brightness of a bleaching plant (paper pulp industry), in a neuro-fuzzy framework, is developed in two phases: subtractive clustering to obtain a set of fuzzy rules and then a fuzzy neural network is trained to optimally tune the membership parameters using backpropagation.
Some problems were encountered related to the variable pure time delay of the process which limited the accuracy of the obtained model. The system may also be time varying.

Another problem comes from the quality of industrial data. It seems that for a deeper foundation of the methodology, more data, with lower sampling interval, should be used, in order to capture the short term dynamics of the process. This means more and better instrumentation. However the shown results illustrate the potentialities of the proposed methodology.

REFERENCES

Chiu, S. (1994). Fuzzy model identification based on cluster estimation. *Journal of Intelligent and Fuzzy Systems*, **2** (3), 267-278.

Caima (1994). TCF bleaching at Caima: past and future. *Caima's Work Group on Best*, Companhia de Celulose do Caima, Internal Report.

Delgado, M., A. F. Gomez, S. Karmeta and F. Martin (1997). A fuzzy clustering-based rapid prototyping for fuzzy rule-based modeling. *IEEE Trans on Fuzzy Systems*, **5** (5), 223-233.

Duarte, B. (1995). *Bleaching plant fuzzy model*. Companhia de Celulose do Caima, Internal Report.

Lin, C.-T. (1995). A neural fuzzy control scheme with structure and parameter learning. *Fuzzy Sets and Systems*, 70, 183-212.

Pedrycz, W. and J. Waletzky (1997). Fuzzy Clustering with Partial Supervision. *IEEE Proc. SMC* part B, **27** (5), 787-795.

Runkler, T.A. (1998). Automatic generation of first order Takagi-Sugeno systems using fuzzy c-elliptotypes clustering. *Journal of Intelligent and Fuzzy Systems*, **6** (4), 435-445.

Wang, Y. and G. Rong (1999). A self-organizing neural-network-based fuzzy systems. *Fuzzy Sets and Systems*, **103** (1), 1-11.

ACKNOWLEDGEMENTS

This work was partially supported by the Portuguese Ministry of Science and Technology (MCT), under the program PRAXIS XXI, contract (Alcine) PRAXIS XXI/EEI/11455/1998.