FUZZY PREDICTIVE CONTROL OF A COLUMN FLOTATION PROCESS

S. Vieira * J. M. C. Sousa * F. O. Durão **

* *Technical University of Lisbon, Instituto Superior Técnico*
Dept. of Mechanical Engineering/ GCAR - IDMEC
Av. Rovisco Pais, 1049-001 Lisbon, Portugal
** *Technical University of Lisbon, Instituto Superior Técnico*
Dept. Mining Engineering/ CVRM
Av. Rovisco Pais, 1049-001 Lisbon, Portugal

Abstract: Column flotation processes are difficult to control because they are multivariable, difficult to model and complex systems. Model predictive control is a model-based control strategy that has been applied to a large number of industrial processes, where a sequence of future control actions is computed by minimizing an objective function. Accurate nonlinear models using soft computing (e.g. fuzzy and neural) techniques are increasingly being used in model based control. In this paper, model predictive control is applied to a column flotation process using a nonlinear fuzzy model. The process has four manipulating variables: feed flow rate, washing water, air and rejected flow rates. The outputs of this model, which are normally used to control the grade and the recovery in the flotation column, are the froth layer height, the bias flow rate and the air holdup in the collection zone. The most important controlled variable is the froth layer height which in this work has a very good performance.

Keywords: Fuzzy model, Model Predictive Control, Branch and bound.

1. INTRODUCTION

Industrial applications are always challenging and have inherent complexity. Column flotation is nowadays an important mineral processing unit. It is a complex multivariable process undergoing several disturbances, such as those originated by changes in feed characteristics and in equipment wearing (Finch and Dobby, 1990). Classic and modern approaches to process control rely on the availability of dynamic models of the process. However, until now it has been difficult to develop a phenomenological dynamic model of the column flotation.

Model Predictive Control (MPC) is a model-based control strategy that has been applied to a large number of industrial processes. Model predictive control uses a process model to predict future process outputs, where a sequence of future control actions is computed by minimizing an objective function (Maciejowski, 2001). Accurate nonlinear models using soft computing (e.g. fuzzy and neural) techniques are increasingly being used in model based control. Both the use of these nonlinear models and the presence of constraints lead to a non-convex optimization problem, which must be solved at each time instant. By formulating the decision problem as a discrete choice problem, the optimization can be efficiently performed by search algorithms like branch and bound (B&B). In this case, the control space is discretized and the problem is reduced to searching the best control action in the space of control actions (Roubos *et al.*, 1999; Mendonça *et al.*, 2004). The discretization of the control space introduces a tradeoff between the number of discrete alternatives and the computational complexity.

The nonlinear fuzzy model, used in the predictive control, is obtained using input-output data from a laboratory column flotation in order to try to achieve a model as accurate as possible. The structure of the model is determined using a regularity criterion (RC) to find, "automatically", the dynamic relations between input-output variables, as proposed in (Sugeno and Yasukawa, 1993). The fuzzy model used are Takagi-Sugeno (TS) fuzzy models. The fuzzy rules are identified using the Gustafson-Kessel fuzzy clustering algorithm (Gustafson and Kessel, 1979). The inicial model is optimized using real-coded genetic algorithms. This combination leaded to the most accurate fuzzy models (Vieira *et al.*, 2004).

The paper is organized as follows. Fuzzy modelling is briefly described in Section 2 and model predictive control is presented in Section 3. A brief description of the column flotation process is presented in Section 4. The proposed control scheme is applied to the control of the column flotation process and some results and comments are presented in Section 5. Finally, some conclusions are drawn in Section 6.

2. FUZZY MODELING

Fuzzy modelling using measures of the process variables, is a tool that allows an approximation of nonlinear systems when there is no prior knowledge about the system or when it is only partially known. Usually, fuzzy modelling follows three steps: structure identification, parameter estimation and model validation.

One of the important advantages of fuzzy models is that they combine numerical accuracy with transparency in the form of linguistic rules. Hence, fuzzy models take an intermediate place between numerical and symbolic models (Babuška, 1998). In computational terms, fuzzy models are flexible mathematical structures that are known to be universal function approximators. Usually the achieved fuzzy model has better performance and accuracy than classical linear models. The system to be identified can be represented as a MIMO nonlinear auto-regressive (NARX) model: $\mathbf{y} = f(\mathbf{x})$, where $\mathbf{x}$ is a state vector obtained from input–output data. In this case, the state vector $\mathbf{x}$ at each time instant k can be obtained from the inputs and outputs of the system, joining them in a vector.

2.1 Structure Identification

In this paper, the significant state variables are chosen using the regularity criterion, as proposed in (Sugeno and Yasukawa, 1993) and applied in (Vieira *et al.*, 2004).

To apply this criterion, the identification data must be divided into two groups, A and B. The regularity criterion is used, e.g., for data handling, which is defined as follows:

$$RC=\left[\sum_{i=1}^{k_A}(y_i^A-y_i^{AB})^2/k_A+\sum_{i=1}^{k_B}(y_i^B-y_i^{BA})^2/k_B\right]/2 \quad (1)$$

where k_A and k_B are the number of data points of the groups A and B, respectively, y_i^A and y_i^B are the output data of the groups A and B, respectively, y^{AB} is the model output for the group A estimated using the data from group B, and y^{BA} is the model output for the group B estimated using the data from group A.

The number of clusters that best suits the data must be determined. The following criterion, as proposed in (Sugeno and Yasukawa, 1993), is used to determine the number of clusters:

$$S(c)=\sum_{k=1}^{N}\sum_{i=1}^{c}(\mu_{ik})^m(\parallel \mathbf{x}_k-v_i \parallel^2-\parallel v_i-\bar{\mathbf{x}} \parallel^2), \quad (2)$$

where N is the number of data to be clustered, c is the number of clusters ($c \geq 2$), $\mathbf{x}_k$ is the k^{th} data point (usually vector), $\bar{\mathbf{x}}$ is the mean value for the inputs, v_i the center of the i^{th} cluster, μ_{ik} is the grade of the k^{th} data point belonging to i^{th} cluster and m is a adjustable weight; usually $m \in [1.5, 3]$. The number of clusters c is increased from two up to the number that gives the minimum value for $S(c)$.

2.2 RC algorithm

Assuming that input and output data are collected using the pilot scale column flotation, structure identification using this methodology generally entails the following algorithm:

(1) Cluster the data using fuzzy c-means with 2 initial clusters;
(2) Compute equation (2)
(3) Increase the number of clusters until equation (2) reach its minimum;
(4) Divide the data set into two groups A and B;
(5) REPEAT for each state in the state vector that does not belong to the inputs of the model,
  (a) Build two models, one using data group A and other using data group B:
  (b) Compute equation (1);
  (c) Put the state with the lowest RC as a new input of the model;
(6) UNTIL RC increases or the end of the state vector is reached.
(7) Select the final inputs;
(8) Using the number of clusters given from equation (2) and the inputs selected by equation (1), build a fuzzy model using GK clustering algorithm.

2.3 Optimal Parameter estimation

One of the techniques that is especially suitable for constrained, nonlinear optimization problems are the *evolutionary computation* techniques (Michalewicz and Fogel, 2002), from which genetic algorithms (GA) is the most common. GA are inspired by the biological process of natural selection, performing selection, crossover and mutation over a population, in order to achieve a global optimum. Instead of searching from general-to-specific hypotheses or from simple-to-complex, genetic algorithms generate successor hypotheses by repeatedly mutating and recombining parts of the best currently known hypotheses. GA are applied to an existing population of individuals, the chromosomes. At each iteration of the genetic process, an evolution is obtained by replacing elements of the population by offsprings of the most fitted elements of that same population. In this way, the best fit individuals have a higher possibility of having their offspring (that represent variations of itself) included in the next generation. The genetic algorithm described in this paper is based on the real-coded genetic algorithm to optimize fuzzy models, proposed in (Setnes and Roubos, 2000).

2.4 Genetic algorithm for fuzzy model optimization

Given the data matrix the structure of the fuzzy rule base derived using the algorithm described in Section 2.2, select the number of generations N_g and the population size L.

(1) Create the initial population based on the derived fuzzy model structure.
(2) Repeat genetic optimization for $t = 1, \ldots, N_g$:
   (a) Select the chromosomes for operation and deletion.
   (b) Create the next generation: operate on the chromosomes selected for operation and substitute the chromosomes selected for deletion by the resulting offspring.
   (c) Evaluate the next generation by computing the fitness for each individual.
(3) Select the best individual (solution) from the final generation.

3. PREDICTIVE CONTROL

Predictive control is a general methodology for solving control problems in the time domain having one common feature: the controller is based on the prediction of the future system behavior by using a process model. Model predictive control is based on the use of an available (nonlinear) model to predict the process outputs at future discrete times over a prediction horizon. With this method, a sequence of future control actions is computed using this model by minimizing a certain objective function.

Usually the *receding horizon principle* is applied, i.e., at each sampling instant the optimization process is repeated with new measurements, and the first control actions obtained are applied to the process. Because of the explicit use of a process model and the optimization approach, MPC can handle multivariable processes with nonlinearities, non-minimum phase behavior or long time delays, and can efficiently deal with constraints (Maciejowski, 2001).

The future plant outputs for a determined *prediction horizon* H_p are predicted at each time instant k by using a model of the process. The predicted output values $\hat{\mathbf{y}}(k+i)$, $i = 1, \ldots, H_p$ depend on the states of the process at the current time k and on the future control signals $\mathbf{u}(k+j)$, $j = 1, \ldots, H_c$, where H_c is the *control horizon*. The control signals change only inside the control horizon, remaining constant afterwards, i.e., $\mathbf{u}(k+j) = \mathbf{u}(k+H_c-1)$, for $j = H_c, \ldots, H_p - 1$.

The sequence of future control signals is obtained by optimizing a cost function which describes the control goal. Let the error for the several outputs be represented as

$$\mathbf{e} = \mathbf{r}(k+i) - \hat{\mathbf{y}}(k+i)\,. \tag{3}$$

For multivariable systems these goals can be represented, for instance by the objective function:

$$J(\mathbf{u}) = \sum_{i=1}^{H_p} \|\mathbf{e}(k+i)\|^2_{\mathbf{Q}} + \sum_{i=1}^{H_c} \|\Delta\mathbf{u}(k+i-1)\|^2_{\mathbf{R}} \tag{4}$$

or some small modifications of it, where $\hat{\mathbf{y}}$ are the predicted process outputs, $\mathbf{r}$ is the reference trajectory, and $\Delta\mathbf{u}$ is the change in the control signals. The first term of (4) accounts for the minimization of the output errors and the second term represents the minimization of the control effort. The term considering the control effort can be given directly by the control actions $\mathbf{u}$, which usually minimizes the energy cost. The matrices $\mathbf{Q}$ and $\mathbf{R}$ determine the weighting between the two terms in the global criterion. In general, these matrices are diagonal in order to simplify the weight attribution.

The performance of MPC depends largely on the used process model. The ability of this model to predict the future process outputs and work in real-time is very important. When a linear time-invariant model is used, and in the absence of constraints, an explicit analytic solution of the problem in (4) can be obtained. When any constraint is violated, but the other two conditions remain, no analytical solution is available. The optimization problem results then in a quadratic problem to be solved at each time instant. This nonlinear optimization problem is convex and can be solved using fast gradient-descent methods with a guaranteed global solution. However, in the most general case both nonlinear models and constraints are present, and the optimization problem results in a non-convex problem.

Optimization methods for non-convex optimization problems can be used when the solution space is discretized, where the problem is transformed into a discrete optimization problem. This allows the application of B&B, which is an efficient tool to deal with this type of optimization problem (Roubos *et al.*, 1999; Mendonça *et al.*, 2004).

3.1 Internal model control

Without disturbances, modeling errors and constraints, the controller yields a time-optimal controller with zero steady-state error. The effects of modeling errors and disturbances can be reduced by using the nonlinear internal model control (IMC) scheme (Economou *et al.*, 1986), given in Fig. 1. The IMC scheme consists

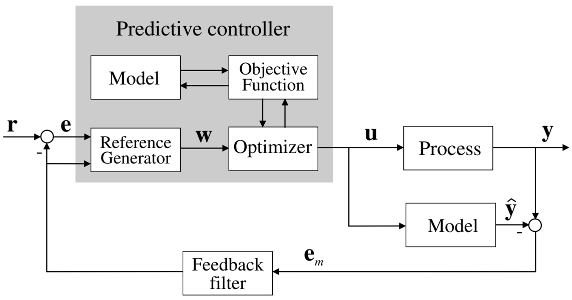

Fig. 1. Internal model control scheme.

of three blocks: the fuzzy model of the process, the controller which is based on this fuzzy model, and a feedback filter. The purpose of the fuzzy model working in parallel with the process is to subtract the effect of the control action from the process output. If the predicted and the measured process outputs are equal, the error between the process output and the model output, $\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}}$, is zero and the controller works in an open-loop configuration. If a disturbance acts on the process output, the feedback signal, $\mathbf{e}$, is equal to the influence of the disturbance and is not affected by the effects of the control action. This signal is simply subtracted from the reference.

3.2 Branch-and-Bound Optimization

The B&B method solves a problem by dividing it into smaller subproblems, using a tree structure. In the space solution, only a small number of possible solutions needs to be enumerated, while the remaining solutions are eliminated because they do not contain an optimal solution. The set of solutions not eliminated is subsequently partitioned into increasingly refined parts (branching) over which lower and upper bounds for the optimal value of the objective function can be determined (bounding). The B&B operations are applied recursively. When the control actions are discretized, the B&B method can be applied to predictive control (Roubos *et al.*, 1999). The B&B optimization technique applied to predictive control has several advantages over other nonlinear optimization methods. First, the global minimum is always found, guaranteeing optimality in the discrete control space. Secondly, the algorithm is not negatively influenced by a poor initialization, as in the case of iterative optimization methods. Finally, the B&B method implicitly deals with constraints. These improve the efficiency of bounding, by restricting the search space and eliminating the control actions that are not valid (do not respect the constraints, for instance).

A general formulation has M_i discrete control actions for each u_i. Without loss of generality, in this paper each input u_i of the system is discretized into M discrete control actions, where M is the same for all the control actions (Mendonça *et al.*, 2004). Therefore, a discrete control action is represented by u_{ij}, with $i = 1, \ldots, m$ and $j = 1, \ldots, M$.

The discrete set Ω containing all the possible control actions is given by:

$$\Omega = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_m \tag{5}$$

where each Ω_i represents the set of all possible discrete control actions for the input u_i: $\Omega_i = \{\omega_{ij} | j = 1, \ldots, M\}$. The number of the total possible discrete control actions S is given by:

$$S = \underbrace{M \times M \times \cdots \times M}_{m \text{ times}} \tag{6}$$

At each time step, S control alternatives can be considered, yielding a maximum of S branches.

Now, let $i = 1, \ldots, H_p$ denote the ith level of the tree ($i = 0$ at initial node) and let j denote the branch corresponding to the control alternative ω_j. The cumulative cost at node *i*, $J^{(i)}$ is given by

$$J^{(i)} = \sum_{\ell=1}^{i} \left(\|\mathbf{e}(k+\ell)\|^2_{\mathbf{Q}} + \|\Delta\mathbf{u}(k+\ell-1)\|^2_{\mathbf{R}} \right) \tag{7}$$

In this B&B formulation, no branching takes place beyond the control horizon ($i > H_c - 1$). Therefore, the control action $\mathbf{u}(k+H_c-1)$ is applied successively until H_p is reached. If only the branching rule would be applied, this would result in an enumerative search, and S^{H_c} possibilities would be tested. Even for a small number of inputs m, small number of discretizations M, and small control horizons, this number can be too large, inducing an enormous computational effort. Thus, the bounding task is fundamental to reduce the number of alternatives. A particular branch j at level i is followed only if the cumulative cost $J^{(i)}$ plus a lower bound on the cost from the level i to the terminal level H_p, denoted $J_L^{(i)}$ is lower than an upper bound of the total cost, denoted J_U. In this paper, we assume that the lower bound is given by the cost associated with the transition $\hat{\mathbf{y}}(k+i) = f(\mathbf{x}(k+i-1), \omega_j)$, which is computed using the cost function (7), and is represented as $J_j^{(i)}(\omega_j)$. The remaining cost from $i+1$ up to H_p is very hard to calculate, and we assume that it is zero. Therefore, the branch condition is the following:

$$J^{(i)} + J_j^{(i)}(\omega_j) < J_U \quad (8)$$

4. APPLICATION TO THE COLUMN FLOTATION PROCESS

4.1 Process

Froth flotation or, shortly, flotation (as froth flotation is the most important of the flotation processes) was introduced in the beginning of the 20^{th} century, and is one of the most versatile separation processes used in mineral processing. Until then, it was not possible to separate several minerals, like most of sulfides, because they exhibit similar density, magnetic susceptibility and conductivity.

The flotation process separates fine solid particles based on physical and chemical properties of their surfaces. Industrially, it is a continuous solid-solid separation process performed in a vessel where a three-phase system is present: solid particles, air bubbles and water. This pulp is previously conditioned with the controlled addition of small quantities of specific chemical reagents to promote the selective formation of aggregates between solid particles of a given composition and air bubbles. Air is continuously injected in the pulp, giving rise to the formation of air bubbles.

Hydrophobic particles adhere, after collision, to the air bubbles, which move upwards to the top of the vessel where they are recovered as the floated product. Hydrophilic particles settle in the pulp, become the non-floated product or underflow. Besides mineral processing, it is used in some other fields, such as solvent extraction and recycling. A flow of air is continuously injected in the medium to transport the particles. After collision with air bubbles, which move upwards to the top of the column, hydrophobic particles adhere to them and will be recovered as the floated product. Hydrophilic particles stay in the vessel, becoming the underflow product, recovered at the bottom of the column.

There is also a shower of water in the top of the froth column, used to ”wash” from the floated product hydrophilic particles that were dragged with the aggregates bubbles-hydrophobic particles.

The work presented in this paper was performed using experimental data of a pilot scale laboratory flotation column of 3.2m high by 80mm of diameter. The flotation column is equipped with variable speed peristaltic pumps (manipulation of underflow and feed flow rates) and control valves (manipulation of air and

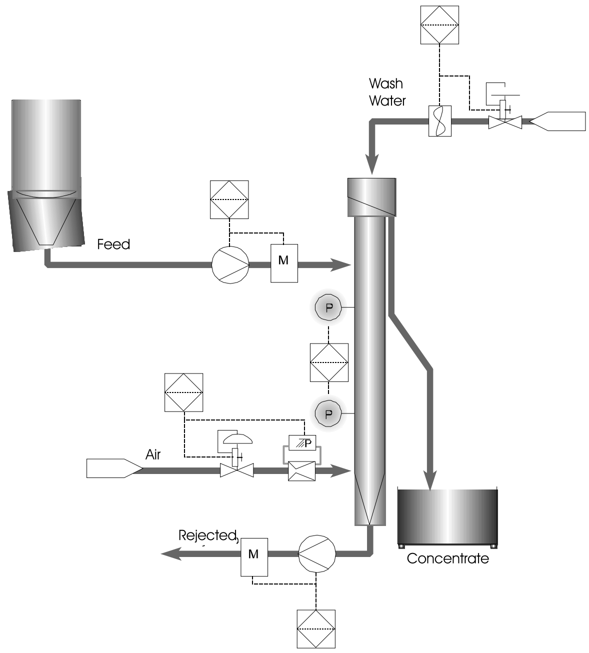

Fig. 2. Column flotation scheme, where M are flow meters and P are pressure sensors.

Table 1. Variance acounted for (VAF) of the obtained fuzzy models.

	H	Q_{bias}	ϵ_c
RC criterion (used as model)	76.6	92.8	88.0
RC + GA (used as real process)	98.6	94.9	90.9

wash water flow rates). The underflow and feed flow rates are measured by electromagnetic flowmeters, the wash water flow rate by a turbine meter and the air flow rate by an orifice plate, see Fig. 2. The collection zone height is inferred by means of a *soft sensor* (Carvalho, 1998) that uses the measurements of two pressure sensors. These measurements are also used to estimate the air holdup in the collection zone. The bias water flow rate is calculated as the difference between feed and overflow flow rates.

At this stage, the system is operated with air and water.

5. RESULTS AND COMMENTS

The application of the described predictive control scheme to the column flotation process aims the stabilization of the internal variables of the process under study.

The column flotation process is too complex, and up to now, no physical model is readily available. For this reason, the simulations were performed using two different fuzzy models, one more accurate (used as the process) than the other (used as the model). The models accuracy and performance were presented in (Vieira *et al.*, 2004). Table 1 presents the accuracy of the fuzzy models, measured with the variance accounted for (VAF). Tables 2 and 3 show the physical range of the process variables, used in the design

Table 2. Range of the controled variables.

	Min.	Max.
H	0	68
Q_{bias}	-10	10
ϵ_c	0	10

Table 3. Range of the manipulated variables.

	Min.	Max.
Q_{air}	150	330
Q_{rej}	60	160
Q_{feed}	70	160
Q_{ww}	30	54

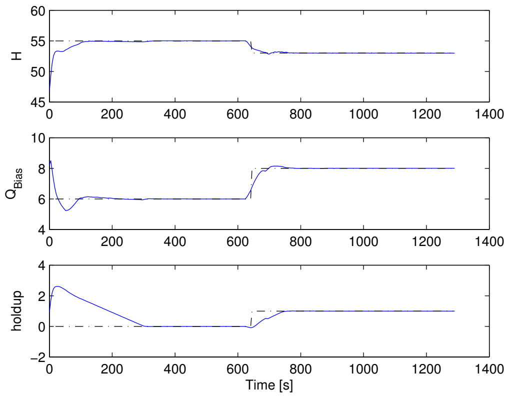

Fig. 3. Model Predictive Control results.

of the predictive controller. From all the controlled variables, the level (H) is the most important in the stabilization process. For this reason, the control of this variable is considered more important than the rest of controlled variables. Both the level (H) and the air holdup (ϵ_c) are strongly influenced by one of the manipulated variables, the air flowrate. The results shown in Fig. 3 were obtained using six discretizations, a control horizon (H_p) of one and a prediction horizon (H_p)of eight.Given the importance of the level control, it is considered that the air holdup may oscillate from is reference between 10 to 25% of his range, as stated in (Carvalho, 1998). The bias flowrate (Q_{bias}) is influenced mainly by the feed flowrate (Q_{feed}) and the rejected flowrate (Q_{rej}).

When a nonlinear model of the process is used, the resulting optimization problem in model predictive control is often non-convex. B&B can be used to search a discretized control space for the optimal solution, requiring a small number of discrete control alternatives. But in this case the number of discretizations of the control actions has a strong influence on the results and it is not possible to control the process with less than four discretizations. This may increase exponentially the computation time.

6. CONCLUSIONS

Model predictive control was applied successfully to column flotation process, which is a very complex, nonlinear and multivariable system. The inherent instability of the system, makes it difficult to control. The results show that the main controlled variable has a good performance and the two other also show a good performance, however there is some work to do, expecting to reduce the settling time especially of the air holdup. The fuzzy model presented in this paper, which presents very good performance, is very important, once it minimizes the prediction error. The predictive controller will be implemented in the pilot plant in the near future.

ACKNOWLEDGMENTS

This work is supported by the project POCTI/ECM /38296/2001-APTDEC, Fundação para a Ciência e Tecnologia, Ministério do Ensino Superior e da Ciência and FEDER, Portugal.

REFERENCES

Babuška, R. (1998). *Fuzzy Modeling for Control*. Kluwer Academic Publishers. Boston.

Carvalho, T. (1998). Fuzzy Control of a Column Flotation Process (in Portuguese). Ph.d. dissertation. Tech. Univ. of Lisbon, IST. Lisbon, Portugal.

Economou, C.G., M. Morari and B. Palsson (1986). Internal model control, extension to nonlinear systems. *Ind. Eng. Chem. Process Des. Dev.* **25**, 403–411.

Finch, J. and J. Dobby (1990). *Column Flotation*. Pergamon Press.

Gustafson, D.E. and W.C. Kessel (1979). Fuzzy clustering with a fuzzy covariance matrix. In: *Proc. of the* 18^{th} *IEEE Conference on Decision and Control*. San Diego, CA, USA. pp. 761–766.

Maciejowski, J. M. (2001). *Predictive Control with Constraints*. Prentice Hall Inc.. New York, USA.

Mendonça, L.F., J. M. Sousa and J.M.G. Sá da Costa (2004). Optimization problems in multivariable fuzzy predictive control. *International Journal of Approximate Reasoning* **36**(3), 199–221.

Michalewicz, Z. and D. B. Fogel (2002). *How to solve it: Modern Heuristics*. third ed.. Springer.

Roubos, J.A., S. Mollov, R. Babuška and H.B. Verbruggen (1999). Fuzzy model-based predictive control using takagi-sugeno models. *International Journal of aproximate Reasoning* **22**, 3–30.

Setnes, M. and J.A. Roubos (2000). Ga-fuzzy modeling and classification: complexity and performance. *IEEE Trans. on Fuzzy Sys.* **8**(5), 509–522.

Sugeno, Michio and Takahiro Yasukawa (1993). A fuzzy-logic-based approach to qualitative modeling. *IEEE Trans. on Fuzzy Sys.* **1**(1), 7–31.

Vieira, S., J. M. Sousa and F. Durão (2004). Combination of fuzzy identification algorithms applied to a column flotation process. In: *Proc. IEEE Int. Conf. on Fuzzy Sys., FUZZ-IEEE'2004*. Budapest, Hungary. pp. 421–426.