INT J COMPUT COMMUN, ISSN 1841-9836
Vol.7 (2012), No. 4 (November), pp. 645-660

A Multimodel Approach for Complex Systems Modeling based on Classification Algorithms

N. Elfelly, J-Y Dieulot, M. Benrejeb, P. Borne

Nesrine Elfelly, Jean-Yves Dieulot,
Mohamed Benrejeb, Pierre Borne
Université des Sciences et Technologies de Lille (USTL),
Ecole Polytechnique de lille,
Ecole Nationale d'Ingénieurs de Tunis -LARA Automatique (ENIT),
Ecole Centrale de Lille (EC Lille)
Laboratoire d'Automatique, Génie Informatique et Signal
Ecole Centrale de Lille, Cité scientifique BP 48
59651 Villeneuve d'Ascq Cedex, France
E-mail:nesrine.elfelly@ed.univ-lille1.fr,
jean-yves.dieulot@polytech-lille.fr,
mohamed.nerejeb@ec-lille.fr,pierre.borne@ec-lille.fr

Abstract: In this paper, a new multimodel approach for complex systems modeling based on classification algorithms is presented. It requires firstly the determination of the model-base. For this, the number of models is selected via a neural network and a rival penalized competitive learning (RPCL), and the operating clusters are identified by using the fuzzy K-means algorithm. The obtained results are then exploited for the parametric identification of the models. The second step consists in validating the proposed model-base by using the adequate method of validity computation. Two examples are presented in this paper which show the efficiency of the proposed approach.
Keywords: complex systems , multimodel , system modeling, classification

1 Introduction

The multimodel approach has been recently developed in several science and engineering domains, with typical applications in the mechanical and chemical engineering areas, with application to modelling, control and/or fault detection e.g. [1–3]. It was introduced as an efficient and powerful method to cope with modelling and control difficulties when complex non linear and/or uncertain processes are concerned. The multimodel approach assumes that it is possible to replace a unique non linear representation by a combination of simpler models thus building a so-called model-base. Usually, each model of this base describes the considered process at a specific operating point. The interaction between the different models of the base through normalized activation functions allows the modelling of the global non-linear and complex system. Therefore, the multimodel approach aims at lowering the system complexity by studying its behavior under specific conditions. The multimodel principle is given in figure 1.

The different models of the base could be of different structures and orders but no model can represent the system in its whole operating domain. The decision unit allows the estimation of the weight of each model and thus the selection of the most relevant models at each time. As for the output unit, controlled by the decision unit, it allows the computation of the multimodel output which is obtained by the contribution of the different models' outputs. In spite of its success in many fields (academic, biomedical, etc), the multimodel approach remains confronted with several difficulties, in particular the determination of the number and parameters of the different models representative of the system and the choice of an adequate method of validities computation used for multimodel output deduction. Last years, many authors [4–7] have been

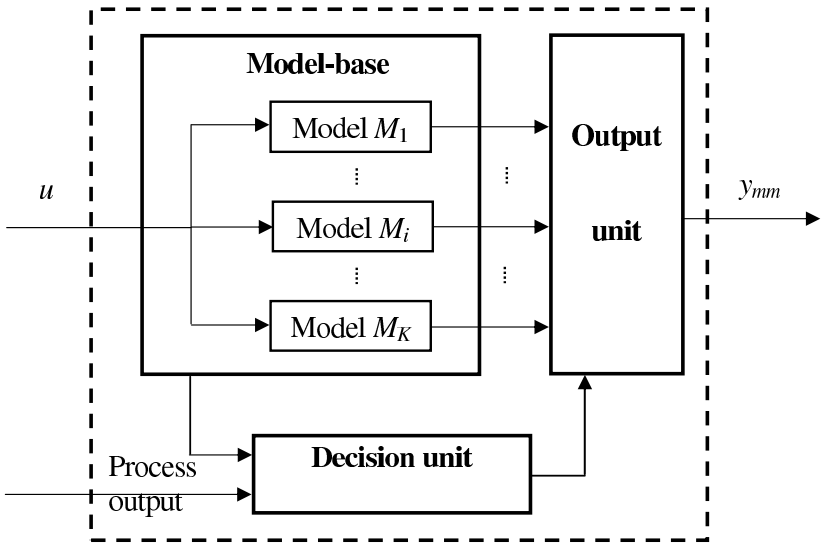

Figure 1: Multimodel approach principle.

interested by this approach. The main differences between the proposed studies are the selected method for models identification and the type of models. Linear models are mainly used, e.g. fuzzy Takagi-Sugeno models [8, 9] which are often obtained using linearization methods. However, the multimodel identification is more difficult to work out when the models of the base should be determined from only input/output data. Some results are given in [10]. In the latter case, the number of models must firstly be determined. Then, the data are classified and the models' parameters are estimated. Last, the validities of the different models are computed according to the adequate method. The issue of the selection of the appropriate method of validities' estimation was discussed in [11]. Many authors propose to apply classification algorithms in order to handle a set of dynamical models. For example, neural networks have been used to represent and control complex systems [7, 12–15]. In another hand, thanks to their ability to classify data and their simplicity, K-means algorithms have proved to be efficient for data clustering e.g. [16, 17]. In short, whereas many architectures using multiple models and neural networks have been proposed, there has not been much work on clustering techniques, based on neural networks and K-means algorithms [18], applied to traditional multimodel representation using only input/output data. The most tedious issues are related to the model-base size and the clustering procedure which aims to the determination of the operating domains. This paper thus proposes a multimodel approach for complex processes modelling based on classification algorithms. The efficiency of the proposed study is illustrated by two examples for which simulations are proposed. The issue of the determination of the number of models in the base is solved through the application of the Rival Penalized Competitive Learning (RPCL), which is an extension of the work in [19] . In the following sections, the different steps of model-base building are detailed knowing that only input/output data are available.

2 Model-base determination using classification algorithms

The proposed approach allows the determination of the number, structure and parameters of the different models of the base. Firstly, a neural network and a rival penalized competitive learning are used for the selection of the adequate number of models. The second step consists in clustering system data by using fuzzy K-means. The classification results are then used for

the parametric identification of the models.

2.1 Determination of the number of models with a Rival Penalized Competitive Learning (RPCL)

The proposed approach allows the construction of the model-base by using two clustering algorithms. The application of this approach requires firstly the determination of the number of models which will be handled by using a Rival Penalized Competitive Learning (RPCL) [20]. Secondly, the application of the fuzzy K-means algorithm for data clustering and then the characterization of the different base-models is carried out by exploiting the clustering results. Finally, the validation of the modeling strategy is considered by the use of an adequate method for validity computation allowing the generation of the multimodel output, compared to the real system output for a different set of inputs.

The models number determination requires experimental data obtained by applying an appropriate input sequence. In order to generate the different operating domains of the process, the measurements must be merged into a set of clusters by the use of a classification algorithm with unsupervised learning. Most existing clustering algorithms [21, 22] do not handle the selection of the appropriate number of clusters, which is, however, essential to the estimation and classification performance in the multimodel approach when no information is available about the operating domains and their number. However, many experimental results have shown that the RPCL algorithm automatically allocates an appropriate number of units for an input data set when they are used for clustering. The selection of the number of models in the multimodel representation requires that the excitation signal must be rich enough (a sine curve for example with the adequate frequency and amplitude added to a random signal) to take into consideration the non-linear aspect of the considered process. For tackling the issue of determination of this number, via input-output data, it is proposed to apply the learning algorithm called RPCL which allows the selection of the adequate number of operating clusters for an input data set. Thus, the extra units are gradually driven far away from the distribution of the data set when the number of units is larger than the real number of clusters in the input data set.

RPCL is an unsupervised learning strategy (proposed by Xu [23] and renewed by Tambe [24]), that automatically determines the optimal number of clusters. The principle underlying RPCL can be considered as an extension of the competitive learning based on Kohonen rule [25]. Its specificity lies in the modification, for each input vector, not only of the winner weights, but also of the weights of its rival (called second winner) so that the rival will be moved or penalized. The rate, at which the rival is penalized, is much smaller than the learning rate, e.g. [26].
Given a competitive learning neural network (Figure 2), i.e. a layer of units with the output u_i of each unit and its weight vector w_i for $i = 1 \dots N$; N is the number of output units, the RPCL algorithm can be described by the following steps.

1. Initialize weight vectors w_i randomly.

2. Take a sample x from a data set D, and for $i = 1 \dots N$, let

$$u_i = \begin{cases} 1 & \text{if} \quad i = c; \\ -1 & \text{if} \quad i = r; \\ 0 & \text{otherwise;} \end{cases} \tag{1}$$

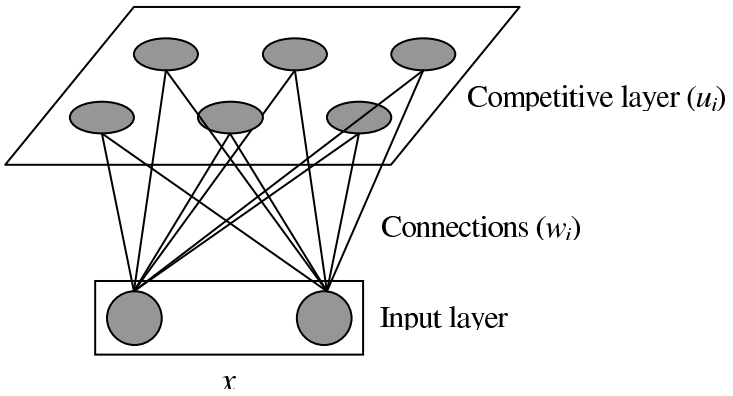

Figure 2: Competitive learning neural network.

with:

$$\gamma_c \|x - w_c\|^2 = \min_j \gamma_j \|x - w_j\|^2; \tag{2a}$$

$$\gamma_r \|x - w_r\|^2 = \min_{j \neq c} \gamma_j \|x - w_j\|^2; \tag{2b}$$

$\| * \|$: Euclidean distance;
c: index of the unit which wins the competition (winner);
w_c: weight vector of the winner;
r: second winner (rival) index;
w_r: weight vector of the rival;
γ_j: conscience factor (relative winning frequency) used to reduce the winning rate of the frequent winners. It is useful to develop a set of equiprobabilistic features or prototypes representing the input data. γ_j is calculated as follows [27]:

$$\gamma_j = \frac{n_j}{\sum_{i=1}^{N} n_i}; \tag{3}$$

where n_j refers to the cumulative number of occurrences the node j has won the competition ($u_j = 1$).

3. Update the weight vectors as follows:

$$w_j(k+1) = w_j(k) + \Delta w_j; \tag{4}$$

with:

$$\Delta w_j = \begin{cases} \alpha_c(k)(x - w_j(k)) & \text{if} \quad u_j = 1; \\ -\alpha_r(k)(x - w_j(k)) & \text{if} \quad u_j = -1; \\ 0 & \text{otherwise;} \end{cases} \tag{5}$$

$0 \leq \alpha_c(k)$ and $0 \leq \alpha_r(k) \leq 1$ are respectively the winner learning rate and the rival de-learning rate. In practice, the rates are fixed small numbers or depend on time (starting from not so small initial values and then reduced to zero in some way) with $\alpha_c(k) \gg \alpha_r(k)$ at each step. Several empirical functions have been proposed for the update of the learning and de-learning rates [27, 28].

4. Repeat steps 2 and 3 until the whole learning process has converged.

Referring to the learning results, only the units enclosed within the data set should be considered and the number of clusters could be deduced as equal to the number of the selected units.

2.2 Identification of the operating clusters

After the selection of the appropriate number of clusters, let us consider the problem of splitting up the measurements to generate the different operating domains from which the base-models will be identified.

Data classification by using fuzzy K-means algorithm

The fuzzy K-means classification algorithm has been chosen for data classification, according to its performance and easy working out.

Fuzzy K-means algorithm

Fuzzy K-means algorithm (developed by Dunn [29] and improved by Bezdek [30]) is a data clustering technique which allows each data point to belong to more than one cluster with different membership degrees (between 0 and 1) and vague or fuzzy boundaries between clusters. The aim of this method is to find an optimal fuzzy K-partition and corresponding prototypes minimizing the following objective function:

$$J_m = \sum_{i=1}^{M}\sum_{j=1}^{K} u_{ij}^m \|x_i - c_j\|^2, \;\; 1 \leq m < \infty; \tag{6}$$

with:
$\| * \|$: any norm expressing the similarity between any measured data and a cluster centre;
m: weighting exponent (real number greater than 1) which is a constant that influences the membership values;
u_{ij}: degree of membership of x_i to cluster j, such as $u_{ij} \in [0,1]$, $\sum_{j=1}^{K} u_{ij} = 1 \;\forall i$ and $0 < \sum_{i=1}^{M} u_{ij} < M \;\forall j$;
x_i: i^{th} data point;
c_j: center vector (node) of the cluster j;
M: number of observations;
K: number of clusters ($2 \leq K < M$).

Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centers c_j [31].
This procedure will stop when $\max\limits_{i,j}\{|u_{ij}(k+1) - u_{ij}(k)|\} < \varepsilon$ and converges to a local minimum of J_m, where ε is a termination criterion belonging to [0,1] and k the iteration step.

The algorithm is composed of the following steps.

1. Initialize the matrix $U = [u_{ij}]$, $U(0)$.
2. At k-step, calculate the centers vectors $c_j; j = 1 \ldots K$:

$$c_j = \frac{\sum_{i=1}^{M} u_{ij}^m \, x_i}{\sum_{i=1}^{M} u_{ij}^m}. \tag{7}$$

3. Update the matrix of membership degrees $U(k)$ according to the new centers positions $U(k+1)$:

$$u_{ij} = \left[\sum_{l=1}^{K} \left(\frac{\|x_i - c_j\|}{\|x_i - c_l\|}\right)^{\frac{2}{m-1}}\right]^{-1}. \tag{8}$$

4. While $\max_{i,j}\{|u_{ij}(k+1) - u_{ij}(k)|\} \geq \varepsilon$, return to step 2.

In our study, the final clustering result is obtained by considering that a given data point belongs to the cluster to which it presents the greatest membership degree.

Parametric identification of the models in the base

The application of the clustering algorithm results in some repartition of the data set. Each cluster is represented by a set of input/output measurements which will be exploited for the identification of the different models in the base. The application of the classification algorithm results in some repartition of the data set. Each cluster is represented by a set of input/output measurements which will be exploited for the identification of the different models in the base. For this, the models orders are first estimated by using the so-called instrumental determinants' ratio-test. This method is mainly based on the conditions concerning a matrix called "information matrix" which contains the input/output measurements [32]. This matrix is described as follows:

$$Q_m = \frac{1}{N_{ob}} \sum_{k=1}^{N_{ob}} \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k-m+1) \\ u(k+m) \end{bmatrix} \begin{bmatrix} y(k+1) & u(k+1) & \cdots & y(k+m) & u(k+m) \end{bmatrix}; \quad (9)$$

where N_{ob} is the number of observations. The Instrumental Determinants' Ratio (IDR) is given by:

$$IDR(m) = \left| \frac{det(Q_m)}{det(Q_{m+1})} \right|. \quad (10)$$

For each value of $m, (m \geq 1)$ the determination procedure of the order consists in building the matrices Q_m and Q_{m+1} and in evaluating the ratio $IDR(m)$; the retained order m is the value for which the ration $IDR(m)$ quickly increases for the first time.
Given those different orders of models, the parametric identification issue consists in calculating the values of the parameters of the corresponding model-equation, given several experimental measures which describe the dynamic behavior of the model. For this, the Recursive Least-Squares method (RLS) [32] is applied to achieve the parameters estimation.

3 Validity computation and validation of the proposed modelling scheme

The steps described in the previous paragraphs allow the design of the model-base. The purpose is to test the efficiency of the proposed modeling. For this, a validation step is worked out for some inputs different from those used for clustering. Then, the multimodel output y_{mm}, computed and compared to the real output of the studied process, is obtained through a fusion of the K models' outputs y_i weighted by their respective validity indexes v_i, as illustrated by the

system 11 and figure 3.

$$y_{mm}(k) = \sum_{i=1}^{K} y_i(k) v_i(k); \tag{11a}$$

$$\sum_{i=1}^{K} v_i(k) = 1. \tag{11b}$$

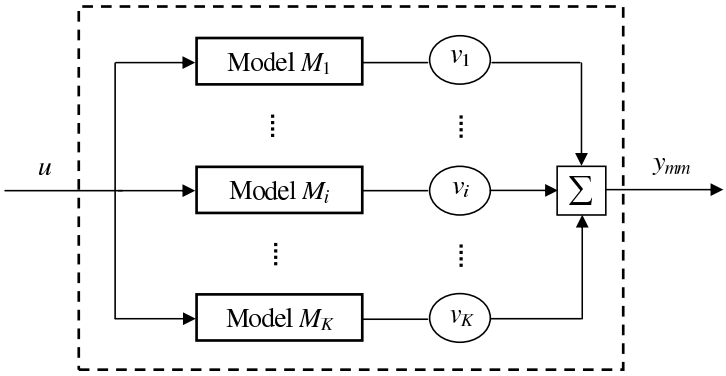

Figure 3: Fusion principle.

The validity index is a real number belonging to the interval $[0, 1]$. It represents the relevance degree of each model calculated at each instant. In the literature, several methods have been proposed to deal with the validity issue. In our study, the residues' approach is adopted for the calculation of validities. This method is based on the distance measurement between the outputs of the process and of the considered model. For example, the residue can be given by the following expression:

$$r_i = |y - y_i|; \qquad i = 1, \ldots, K; \tag{12}$$

with:
y: process output;
y_i: output of the model M_i.

If this residue value is equal to zero, the corresponding model M_i represents perfectly the process at that time; if not, the model represents the process partially.
Between the methods proposed for the calculation of validities [4, 33], only the simple and the reinforced validities approaches are here considered. In general, the expression of the validities is given by:

$$v_i = 1 - r\prime_i \,; \tag{13}$$

where $r\prime_i$ represent the normalized residues and are given by:

$$r\prime_i = \frac{r_i}{\sum_{j=1}^{K} r_j} \,. \tag{14}$$

Simple validities: the normalized simple validities v_i^{simp} are defined so that their sum must be equal to 1 at each time:

$$v_i^{simp} = \frac{v_i}{K - 1} \,. \tag{15}$$

Reinforced validities for this type of validities, the reinforcement expression $v_i^{\prime renf}$ is introduced as:

$$v_i^{\prime renf} = v_i \prod_{j=1, j\neq i}^{K} (1 - v_j)\,. \tag{16}$$

The normalized reinforced validities v_i^{renf} could be written as follows:

$$v_i^{renf} = \frac{v_i^{\prime renf}}{\sum_{j=1}^{K} v_j^{\prime renf}}\,. \tag{17}$$

The comparative study between the two considered validities [11] has shown that the selection of the suitable approach depends on clustering results i.e. the clusters structure and repartition. In fact, when there are important variations in the same cluster and when an overlapping between clusters occurs, it is worth to use the simple validities' method since it takes account of different models' outputs referring to the expression 15. In this case, no model could represent ideally the process at any time. But when the clusters present very few variations and are well separated, the reinforced validities' method is better adapted. Thanks to the reinforcement expression 16, the application of this method promotes the contribution of the most dominant model which represents at best the process behavior.

Validation of the proposed modeling scheme

Once the appropriate method of validity computation selected, the validation of the global modeling scheme is carried out through a comparison between the real and the multimodel outputs for different input sequences.

4 Simulation examples

4.1 Modelling of a mechanical manipulator

In order to underline the interest and the efficiency of the proposed approach, a first example of nonlinear system is considered, which consists of a two-link manipulator (Figure 4) [34, 35] that can be described by the following equations:

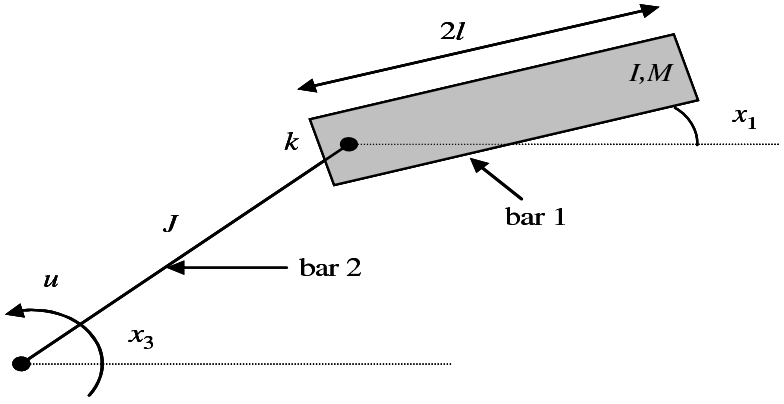

Figure 4: Mechanical system (simulation example).

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = \frac{-Mgl}{I} sin(x_1) - \frac{Mgl}{I}(x_1 - x_3), \\ \dot{x}_3 = x_4, \\ \dot{x}_4 = \frac{k}{J}(x_1 - x_3) + \frac{1}{J}u. \end{cases} \tag{18}$$

The considered system is composed of two rotating bars, where:

- x_1, x_2: angular position and angular velocity of bar 1;
- x_3, x_4: angular position and angular velocity of bar 2;
- u: torque applied to bar 2;
- $g = 9.8\ m \cdot s^{-2}$: gravity constant;
- $I = 1\ kg \cdot m^2$: moment of inertia of bar 1;
- $J = 1\ kg \cdot m^2$: moment of inertia of bar 2;
- $l = 1\ m$: half of the length of bar 1;
- $M = 1\ kg$: mass of bar 1;
- $k = 1\ Nm \cdot rad^{-1}$: elastic rigidity at the link between bars 1-2.

The normal form of the nonlinear model of the system can be written as follows:

$$\begin{cases} \dot{z}_1 = z_2, \\ \dot{z}_2 = z_3, \\ \dot{z}_3 = z_4, \\ \dot{z}_4 = a(z) + b(z)u, \end{cases} \tag{19}$$

where $a(z) = \frac{Mgl}{I} sin(z_1) z_2^2 - \frac{Mgl}{I} cos(z_1) z_3 - (\frac{k}{I} + \frac{k}{I}) z_3 - \frac{kMgl}{IJ} sin(z_1)$ and $b(z) = \frac{k}{IJ}$.

The variables z_i are related to the variables x_i through the following equations:

$$\begin{cases} x_1 = z_1, \\ x_2 = z_2, \\ x_3 = z_1 + \frac{1}{k} z_3 + \frac{Mgl}{k} sin(z_1), \\ x_4 = z_2 + \frac{1}{k} z_4 + \frac{Mgl}{k} cos(z_1) z_2. \end{cases} \tag{20}$$

In the remainder of the study, u will be considered as the input and x_1 as the output of the system. First, the system is excited by an adequate signal $u(k)$ in order to collect the measurements $x_1(k)$ and $x_1(k-1)$ at different instants. These numerical data are used for the determination of the appropriate number of operating clusters by using a neural network and the RPCL algorithm. Figure 5, which gives the results of the learning procedure, shows that by considering six units in the input layer, two centres move away from the observation space, which permits to conclude that the adequate number of clusters can be chosen equal to four.

Then, the fuzzy K-means algorithm is carried out in order to select the different operating clusters (Figure 6).

Referring to each of the four data sets relative to the clusters resulting from the implementation of the proposed classification algorithm, the orders and the parameters of the transfer

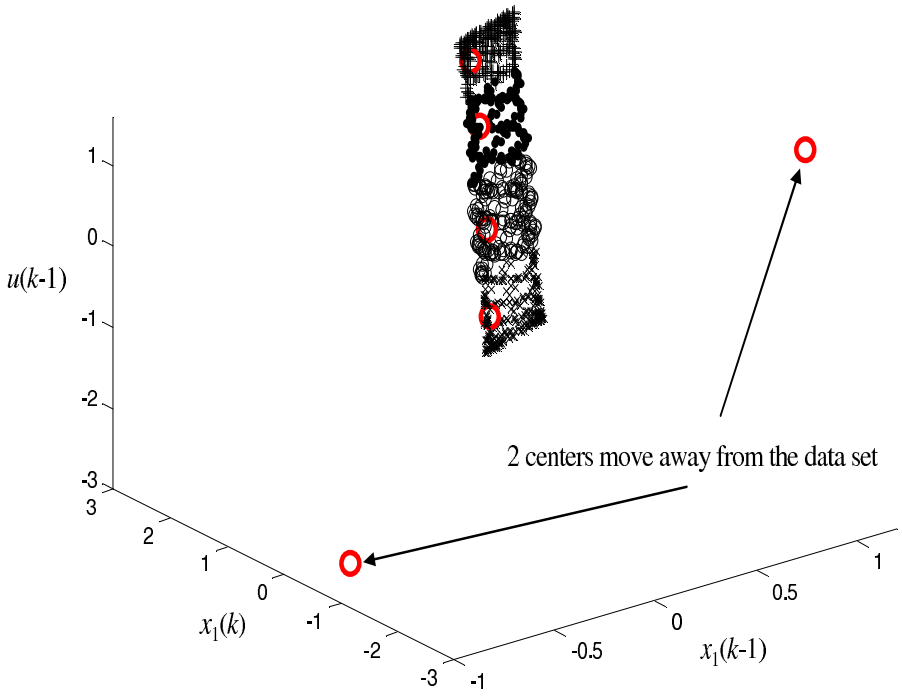

Figure 5: Determination of the number of clusters (N=6).

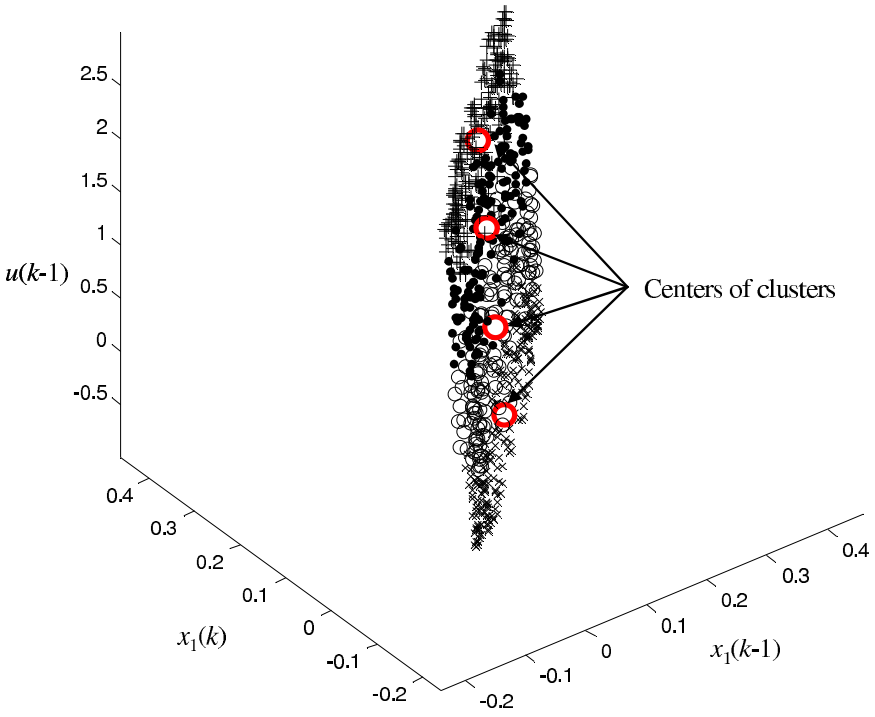

Figure 6: Classification results.

functions relative to the four models of the base are estimated by using respectively the IDR and the RLS methods.

The study presented in [11,19] leads to the selection of the reinforced validities method since the classification results (Figure 6) present well-separated clusters with few variations. In order

to give prominence to the capacity of the identified models to reproduce the operating system in different domains, many input sequences have been considered, two of which are presented in this paper and are described by the following equations:

$$u_1(k) = sin(0.5k). \tag{21}$$

$$u_2(k) = 1 + 0.5sin(0.6k). \tag{22}$$

Referring to the designed model-base, the multimodel output is computed for each input sequence (by fusion of the models' outputs weighted by the reinforced validities) and is then compared to the real system output. The results are given in Figure 7 and Figure 8 where x_1 is the real output, and x_{1mm} the multimodel output obtained by using the fuzzy K-means algorithm.

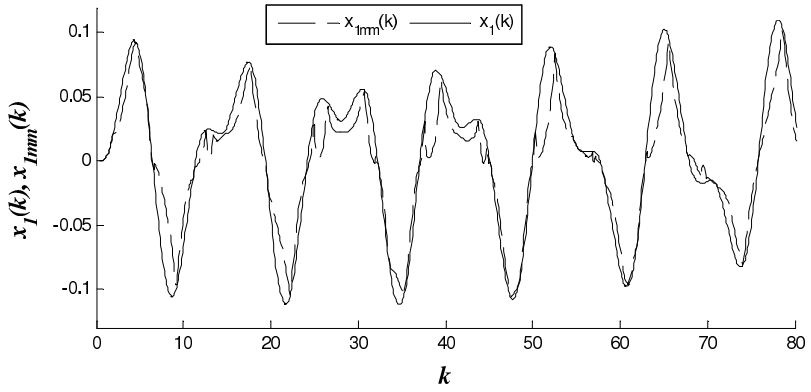

Figure 7: Real and multimodel outputs for the mechanical manipulator (input sequence u_1).

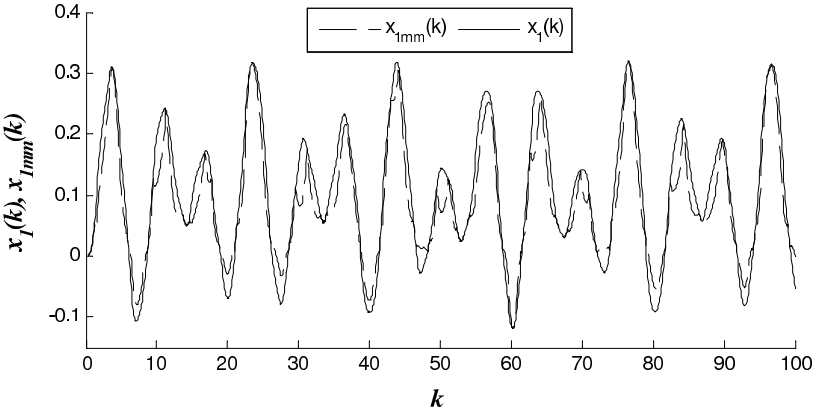

Figure 8: Real and multimodel outputs for the mechanical manipulator (input sequence u_2).

Introducing the NRMSE (Normalized Root Mean Square Error):

$$NRMSE = \frac{\sqrt{\frac{\sum_{i=1}^{N}(y_{mm}-y)^2}{N}}}{y_{max} - y_{min}}, \tag{23}$$

where y_{mm}, y are the multimodel and process output and N is the number of samples; one obtains an accuracy of $NRMSE = 0.001$. A fair comparison was made in a previous paper [39] with an

academic system which was controlled in [6] with a multimodel method where 64 models were proposed instead of only 4 for the method proposed in this paper. The results where only 3 times better for the method in [39] which needed 16 times more models and complicated workout and tuning, but both methods outperformed the results.

It can be noticed that the implementation of the proposed multimodel approach allows an acceptable modelling of the studied system respective to its complexity. In fact, by considering the first input sequence u_1, the multimodel output follows the real output and the error between the two signals shaves over the time. Besides, even with a more complicated system output (obtained by considering the input sequence u_2) with important oscillations and for a longer time interval, the results given by the proposed modelling approach (Figure 8) are satisfactory.

4.2 Modeling of a bioreactor

This second example shows the relevance of our method compared to a "black-box" multimodel technique with a large model base as provided in the work by [6], illustrated on the same benchmark, a bioreactor model. This model has been used in several nonlinear modeling or control issues [6, 36, 37]. Substrate is fed continuously with a constant feedrate F to the well-mixed bioreactor which has a constant volume V. Microbial growth follows a Contois model [38], the microorganisms and substrate concentration, respectively x_1 and x_2, are supposed to be small. The discrete-time mass-balance equations are derived as follows:

$$\begin{cases} x_1(k+1) = x_1(k) + 0.5\frac{x_1x_2}{x_1+x_2} - 0.5u(k)x_1(k), \\ x_2(k+1) = x_2(k) - 0.5\frac{x_1x_2}{x_1+x_2} - 0.5u(k)x_2(k) + 0.05u(k), \\ y(k) = x_1(k) \end{cases} \tag{24}$$

where $y(k)$ is the model output, $u(k) = F/V$ is the dilution rate, normalized to 1. As in the previous example, it is assumed that no a priori information is given on the possible (real) model structure and the system is adequately excited to generate a set of input/output measures which will allow to describe the whole process dynamics. As was explained before, the rival penalized competitive learning algorithm is applied, and the number of models was found to be equal to 10. Experimental measurements are then classified by using the fuzzy k-means algorithm. Based on the learning results, where small overlapping between clusters was found, the reinforced validities were embedded in the multimodel structure. In terms of modelling accuracy, the obtained results were found to be quite similar to those given in [6] where 196 models were needed to represent the system dynamics . This enlightens the interest of the proposed approach as a high number of models yields a considerable computational burden. Figure 9 provides the real output y and the multimodel output y_{mm} by considering the following input sequence: $u(k) = 0.5 + 0.1sin(2k)$. The results are confirmed through another input sequence u_2 shown on figure 10 and figure 11 shows a good agreement between multimodel and real outputs.

5 Conclusion

In this paper, a new approach for complex systems modelling is proposed and an experimental validation is presented. This approach is applicable when dealing with complex, strongly nonlinear and uncertain systems. It allows the determination of the base of models, which are representative of the system in different operating domains, by using two classification algorithms and two methods of structural and parametric identification. The issue of selection of the models' number is solved by using a neural network and a Rival Penalized Competitive Learning (RPCL). Once this number determined, the data collected on the system are then clustered by using

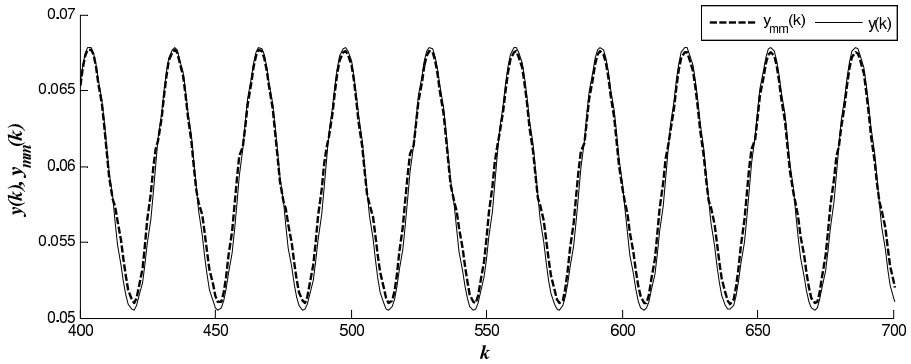

Figure 9: Real and multimodel outputs for the bioreactor model (input sequence u_1).

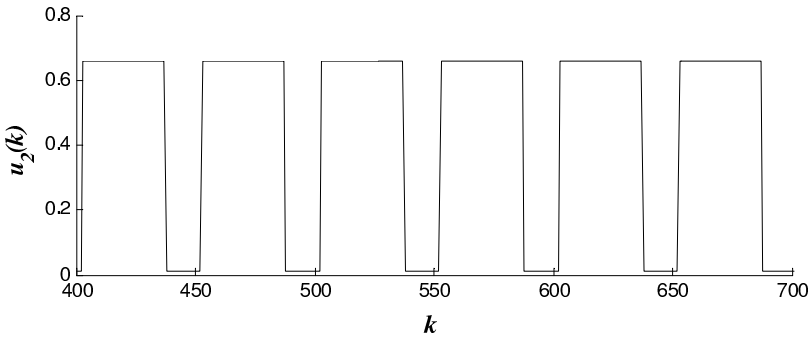

Figure 10: Bioreactor input sequence u_2.

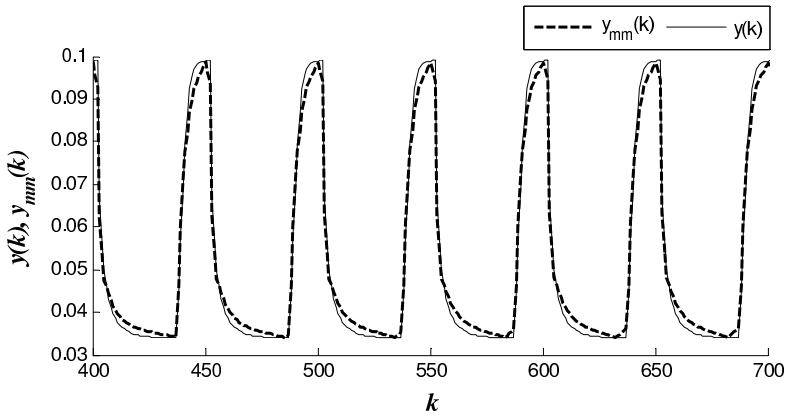

Figure 11: Real and multimodel outputs for the bioreactor model (input sequence u_2).

the fuzzy K-means algorithm. The classification results are exploited for the identification of models. Finally, a validation procedure is worked out in order to demonstrate the ability of the proposed modelling structure to reproduce the system response in different operating domains. The proposed approach has been implemented and applied for a complex mechanical system and for a bioreactor. The obtained results seem to be interesting and prove the efficiency of the proposed modelling strategy.

Bibliography

[1] Alavandar S., Nigam M.J., Neuro-Fuzzy based Approach for Inverse Kinematics Solution of Industrial Robot Manipulators, *INT J COMPUT COMMUN*, 3(3):224-234, 2008.

[2] J. M. Böling and D. E. Seborg and J. P. Hespanha, Multi-model adaptive control of a simulated pH neutralization process, *Control Engineering Practice*, Vol. 15, pp. 663-672, 2007.

[3] M. Rodrigues and D. Theilliol and M. Adam-Medina and D. Sauter A fault detection and isolation scheme for industrial systems based on multiple operating models, *Control Engineering Practice*,16,225-239,2008.

[4] F. Delmotte, L. Dubois, P. Borne, A General Scheme for Multi-Model Controller using Trust, *Mathematics and Computers in Simulation*, Vol. 41, pp. 173-186, 1996.

[5] T. A. Johansen, B. A. Foss, Editorial: Multiple model approaches to modelling and control, *International Journal of Control*, Vol. 72, pp. 575, 1999.

[6] J. Cho and J. C. Principe and D. Erdogmus and M. A. Motter, Quasi-sliding mode control strategy based on multiple-linear models, *Neurocomputing*, 70, 960-974,2007.

[7] I. S. Baruch and R. B. Lopez and J-L. Olivares and J-M. Flores, A fuzzy-neural multi-model for nonlinear systems identification and control, *Fuzzy sets and systems*, 159, 2650-2667, 2008.

[8] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modelling and control, *IEEE Transactions on Systems Man and Cybernetics*, Vol. 15, pp. 116-132, 1985.

[9] M. Benrejeb, D. Soudani, A. Sakly, P. Borne, New discrete TSK fuzzy systems characterization and stability domain, *INT J COMPUT COMMUN*, 1(4):9-19, 2006.

[10] P. Borne and M. Benrejeb, On the Representation and the Stability Study of Large Scale Systems, *INT J COMPUT COMMUN*, 3(S):55-66, 2008.

[11] N. Elfelly and J-Y. Dieulot and P. Borne, A neural approach of multimodel representation of complex processes, *INT J COMPUT COMMUN*,3(2):149-160, 2008.

[12] M. Ronen and Y. Shabtai and H. Guterman, Hybrid model building methodology using unsupervised fuzzy clustering and supervised neural networks, *Biotechnology and Bioengineering*, 77, 420-429, 2002.

[13] W. Yu, Multiple recurrent neural networks for stable adaptive control, *Neurocomputing*, 70, 430-444, 2006.

[14] Y. Fu and T. Chai, Nonlinear multivariable adaptive control using multiple models and neural networks, *Automatica*, 43, 1101-1110, 2007.

[15] G.D. Manioudakis and E.N. Demiris and S.D. Likothanassis, A self-organized neural network based on the multi-model partitioning theory, *Neurocomputing*, 37, 1-29, 2001.

[16] D. Dembélé and P. Kastner, Fuzzy C-means method for clustering microarray data, *Bioinformatics*,19, 973-980, 2003.

[17] P. Hore and L.O. Hall and D.B. Goldgof and W. Cheng, Online Fuzzy C Means, *Annual Meeting of the North American Fuzzy Information Processing Society, NAFIPS 2008*,1-5, 2008.

[18] Z.K. Xue and S.Y. Li, Multi-model modelling and predictive control based on local model networks, *Control and Intelligent Systems*, 34, 105-112, 2006.

[19] N. Elfelly and J.-Y. Dieulot and M. Benrejeb and P. Borne, A new approach for multimodel identification of complex systems based on both neural and fuzzy clustering algorithms, *Engineering Applications of Artificial Intelligence*, 23, 1064-1071, 2010.

[20] L. Xu and A. Krzyzak and E. Oja, Unsupervised and supervised classification by rival penalized competitive learning, *Pattern Recognition*, 11, 496-499, 1992.

[21] A.K. Jain and R.C. Dubes, Algorithms for Clustering Data, *Prentice-Hall Inc., Upper Saddle River, NJ*, 1988.

[22] B. Mirkin, Mathematical Classification and Clustering, *Kluwer Academic Press, Boston-Dordrecht*, 1996.

[23] L. Xu and A. Krzyzak and E. Oja, Rival penalized competitive learning for cluster analysis RBF and curve detection, *IEEE Transactions on Neural Networks*,4, 636-649,1993.

[24] S.S. Tambe and B.D. Kulkarni and P.B. Deshpande, Elements of artificial neural networks with selected application on chemical engineering, and chemical and biological sciences, *Simulation & Advanced Controls Inc., Louisville-KY, USA*, 1996.

[25] T. Kohonen, The self-organizing map, *IEEE Proceedings*, 78, 1464-1480, 1990.

[26] P. Borne and M. Benrejeb and J. Haggège, Les réseaux de neurones, *Editions Technip, Paris, France*, 2007.

[27] T. Murlidharan Nair and C. L. Zheng and J. Lynn Fink and R. O. Stuart and M. Gribskov, Rival Penalized Competitive Learning (RPCL): a topology-determining algorithm for analyzing gene expression data, *Computational Biology and Chemistry*, 27, 565-574, 2003.

[28] I. King and L. Xu and L. Chan, Using rival penalized competitive clustering for feature indexing in Hong Kong's textile and fashion image database, *Proceedings of the IEEE International Joint Conference on Neural Networks*, 1, 237-240, 1998.

[29] J. C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, *Journal of Cybernetics*, 3, 32-57, 1973.

[30] J.C. Bezdek, Pattern recognition with fuzzy objective function algorithms, *Plenum Press, New York*, 1981.

[31] S. Nascimento and B. Mirkin and F. Moura-Pires, A fuzzy clustering model of data and fuzzy C-means, *The Ninth IEEE International Conference on Fuzzy Systems*, 1, 302-307, 2000.

[32] R. Ben Abdennour and P. Borne and M. Ksouri and F. M'sahli, Identification et commande numérique des procédés industriels, *Editions Technip, Paris, France*, 2001.

[33] D. J. Leith, W. E. Leithead, Analytic framework for blended multiple model systems using linear local models, *International Journal of Control*, Vol. 72, pp. 605-619, 1999.

[34] J.V. Salcedo and M. Martínez, Design of PDC fuzzy controllers under persistent disturbances and application in mechanical systems, *Advances in Engineering Software*, 39, 937-946, 2008.

[35] C.-W. Chen, Stability conditions of fuzzy systems and its application to structural and mechanical systems, *Advances in Engineering Software*, 37, 624-629, 2006.

[36] J. P. Gauthier and H. Hammouri and S. Othman, A simple observer for nonlinear systems applications to bioreactors, *IEEE Transactions on Automatic Control*, 37, 875-880, 1992.

[37] G. Bastin and D. Dochain, On-line estimation and adaptive control of bioreactors, *Elsevier, Amsterdam*, 1980.

[38] D. Contois, Kinetics of bacterial growth relationship between population density and specific growth rate of continuous cultures, *Journal of Genetic Macrobiol*, 21, 40-50, 1959.

[39] N. Elfelly and J.-Y. Dieulot and M. Benrejeb and P. Borne, Multimodel control design using unsupervised classifiers, *Studies in Informatics and Control*, ISSN 1220-1766, vol. 21 (1), pp. 101-108, 2012.