A New Approach for Online T-S Fuzzy Identification and Model Predictive Control of Nonlinear Systems

Saeid Rastegar, Rui Araújo, and Jérôme Mendes
Institute for Systems and Robotics (ISR-UC), and
Department of Electrical and Computer Engineering (DEEC-UC),
University of Coimbra, Pólo II, PT-3030-290 Coimbra
srastegar@isr.uc.pt, rui@isr.uc.pt, jermendes@isr.uc.pt

Abstract
This paper proposes a new unsupervised fuzzy clustering algorithm (NUFCA) to construct a novel online evolving Takagi-Sugeno (T-S) fuzzy model identification method and an adaptive predictive process control methodology. The proposed system identification approach consists of two main steps: antecedent T-S fuzzy model parameters identification and consequent parameters identification. The NUFCA combines the K-nearest neighbour and fuzzy C-means methods into a fuzzy modelling method for partitioning of the input-output data and identifying the antecedent parameters of the fuzzy system; then the recursive least squares (RLS) method is exploited to obtain lization typel consequent parameters and to construct a method for on-line fuzzy model identification. The integration of the proposed adaptive identification method with the generalized predictive control (GPC) results in an effective adaptive predictive fuzzy control methodology. For better demonstration of the robustness and efficiency of the proposed methodology, it is applied to the identification of a model for the estimation of the flour concentration in the effluent of a real-world wastewater treatment plant (WWTP); and to control a simulated continuous stirred tank reactor (CSTR), and a real experimental setup composed of two coupled DC motors. The results show that the developed evolving T-S fuzzy model methodology can identify nonlinear systems satisfactorily and can be successfully used for a prediction model of the process for the generalized predictive controller. It is also shown that the algorithm is robust to changes in the initial parameters, and to unexpected disturbances.

Keywords: Adaptive fuzzy model identification, unsupervised fuzzy clustering algorithm, recursive least squares, adaptive predictive control.

1. Introduction

System identification techniques for both data-driven soft sensors (DDSS) and model based predictive control (MPC) play a vital role in the development of any control or monitoring system. A common underlying assumption of methodologies to address these DDSS and MPC problems is their assumption of the knowledge of an accurate model of the process to be identified/controlled. This assumption may cause important problems, because many complex plants are difficult to be mathematically modelled based on physical laws or have large uncertainties and strong nonlinearities. Several types approaches to modelling nonlinear plants can be considered to be used in DDSSs and MPCs. Among them, fuzzy modelling (Zadeh, 1965) has become an active research area because of the capability of handling perceptual uncertainties, such as vagueness and ambiguity involved in the interpretation of a real system. Also, it has shown excellent ability when describing nonlinear systems (Wang & Mendel, 1992; Kosko, 1994), in particular with the Takagi-Sugeno (T-S) fuzzy models (Takagi & Sugeno, 1985). The main task to accomplish a T-S fuzzy model is to run a structure identification procedure, which is concerned with the local dynamics associated to each fuzzy implication (rule), typically described by a local linear system model. It has been proved that Takagi-Sugeno fuzzy models are universal approximators for any smooth nonlinear system (Buckley, 1992; Fantuzzi & Rovatti, 1996).

Like in many other models where optimization algorithms can be used to find parameters, in T-S fuzzy models, optimization algorithms can be used to find the fuzzy consequent parameters. For example, Algorithms such as neural networks (NNs) learning algorithms and genetic algorithms (GAs) have shown a good adaptation to search for optimal T-S fuzzy model parameters (Chen, 2014; Mendes et al., 2012). In (Cazarez-Castro et al., 2010) a hybrid architecture, which combines Type-1 or Type-2 fuzzy logic system and genetic algorithms for the optimization of the membership function parameters is presented. In (Kayadelen, 2011) the potential of genetic expression programming and an adaptive neuro-fuzzy computing paradigm are studied to forecast the safety factor
for liquefaction of soils. Hung & Lin (2012) developed a novel evolutionary algorithm named the partial solutions consideration based a self-adaptive evolutionary algorithm (PSC-SEA) to adjust the parameters of a neuro-fuzzy network that was investigated in other paper by Wang et al. (2012). In (Han et al., 2012) a self-organizing radial basis function neural network model predictive control method is proposed for controlling the dissolved oxygen concentration in a wastewater treatment process (WWTP).

The performance of a control methodology can be affected by the lack of knowledge about the adequate input variables and respective time delays. Otherwise, the method must be robust enough to address the main important concerns in a control design. The methods proposed in (Cazarez-Castro et al., 2010; Kayadelen, 2011; Hung & Lin, 2012; Han et al., 2012) have the limitation of not being able to perform automatic selection of variables: pre-selection is performed. Another key issue is that, in most cases, the collected dataset used in offline methods is limited, and the estimated T-S fuzzy model may not provide adequate accuracy in important parts, or the whole, operating areas of the plant. Moreover, the behaviour and model of the plant may be changing over time. In this case, an online learning strategy would be desired. Online learning of a T-S fuzzy model is based on the assumption that the model structure and parameters evolve gradually. In online T-S model learning, consequent parameters of the model are often recursively estimated (P. Angelov & D. Filev, 2004; Liquan et al., 2012). This work will follow an online learning strategy for the T-S fuzzy modelling.

With respect to computational time, the performances of a T-S fuzzy model depend on its complexity (Number of fuzzy rules), on the type of membership functions, on the number of antecedent variables, and on the consequent regressors. For example, an approach for learning T-S fuzzy models was proposed by (Mendes et al., 2012): a hierarchical evolutionary approach with five levels to optimize the parameters of T-S fuzzy systems is introduced. GAs are usually initialized with random population elements. This sort of approach increases the tuning/search difficulty of the GA, since a set of totally random initial individuals of a population can lead to a very exhausting optimality search, requiring more iterations to attain convergence.

The main motivations of this work are: (1) Introduction of a new unsupervised fuzzy clustering algorithm (NUFCA) to construct the T-S fuzzy model; And (2) Combining NUFCA and a recursive least squares (RLS) method both to construct initial consequent parameters of the T-S model in offline mode, and also as an adaptive approach for online learning/updating, respectively; And (3) The integration of the T-S fuzzy learned by the online identification methodology into an adaptive fuzzy GPC controller (AFGPC). The resulting proposed offline and online identification and adaptive control methodology can deal with non-linear plants, time-varying processes, disturbances or varying operating regions and parameters of the model. The NUFCA will be used to obtain initial T-S fuzzy model parameters such as the value for the number of rules, a set of antecedent membership functions, and a set of rules. Two steps are involved for each candidate number of clusters. In the first step the algorithm finds initial clustering and respective centers. In the second step, the quality of clustering is evaluated using the fuzzy clustering validation index with the goal of searching for the optimal clustering solution. The NUFCA contains two main advantages when comparing with the FCM: (i) Unlike the FCM which needs initial clusters centers and initial clusters numbers, the algorithm proposed in this work does not need any initial estimate; (ii) During performance of the proposed clustering algorithm, a fuzzy clustering validity index is used to evaluate the results of clustering.

To validate the performance and effectiveness of the proposed algorithm, it is applied on an identification problem, and on two control problems. First, a non-linear system identification application problem in the estimation of the flour concentration in the effluent of a real-world wastewater treatment plant is analysed and quantitatively compared with two adaptive approaches: a recursive partial least squares (RPLS) (Dayal & MacGregor, 1997), and a recently proposed incremental local learning soft sensing algorithm (ILLSA) for adaptive soft sensors (Kadlec & Gabrys, 2011). Furthermore, the results are compared with the results of two other methods: a new fuzzy c-regression model algorithm (NFCRMCA) (Li et al., 2009), and a hierarchical genetic approach (HGA) for learning T-S fuzzy models proposed in (Mendes et al., 2012). Then, the performance of the proposed adaptive predictive fuzzy identification and control methodology is demonstrated on two setups: a simulated CSTR plant, and a real-world experimental setup composed of two coupled DC motors. In both the simulated CSTR plant and the two coupled DC motors, the proposed identification performance is illustrated and quantitatively compared with the aforementioned methods. To better demonstrate the efficiency of the NUFCA, also results of experiments regarding the combination of the NUFCA with other methods are presented. Moreover, a comparison between performance of the AFGPC and the classical GPC (Camacho & Bordons, 1998) is exhibited and discussed. The paper is organized as follows. Section 2 presents a description about the nonlinear system modeling methodology based on the T-S fuzzy model, and on the fuzzy C-means clustering algorithm. Section 3 presents the RLS method with adaptive directional forgetting. Section 4 proposes the NUFCA clustering algorithm and introduces the new T-S fuzzy model identification algorithm. In Section 5 a brief overview of the GPC is presented. In Section 6, results in both input variables data classification and identification-control of a plant are presented and analysed. Finally, Section 7 makes concluding remarks.
2. T-S Fuzzy Models Based on Fuzzy C-Means Clustering

This Section presents a nonlinear system modelling methodology based on the T-S fuzzy model, and on Fuzzy C-Means Clustering. Specifically, as an initialization method, the fuzzy C-means clustering algorithm (Celikyilmaz & Trksen, 2009; Dovžan & Škrjanc, 2011; Su et al., 2012) is employed to learn the antecedent parameters of the T-S fuzzy model from data on the T-S fuzzy model learning methodology.

2.1. Modelling Using T-S Fuzzy Models

Takagi-Sugeno fuzzy models with simplified linear rule consequents are universal approximators capable of approximating any continuous nonlinear system (Ying, 1997). For more details about T-S fuzzy models, references (Takagi & Sugeno, 1985; Wang, 1997), are recommended.

With a T-S fuzzy model, the global operation of the nonlinear system can be accurately approximated into several local affine models. In general, a nonlinear system can be described by a T-S fuzzy model defined by the following fuzzy rules:

\[ R_i : \text{IF } x_1(k) \text{ is } A_{i1} \text{ and } \ldots \text{ and } x_N(k) \text{ is } A_{iN} \text{ THEN } y(k) = y_i(k) = \theta_{i1} x_1(k) + \ldots + \theta_{iN} x_N(k), \quad i = 1, 2, \ldots, c, \]

where \( R_i \) (\( i = 1, 2, \ldots, c \)) represents the \( i \)-th fuzzy rule, \( c \) is the number of rules, \( x_1(k), \ldots, x_N(k) \) are the input variables of the T-S fuzzy system. \( A_{ij} \) (\( j = 1, 2, \ldots, N \)) are linguistic terms characterized by fuzzy membership functions \( \mu_{A_{ij}}(x_j) \) which describe the local operating regions of the plant. \( \theta_{i1}, \ldots, \theta_{iN} \) are model parameters of \( y_i(k) \).

From (1), the model output \( y(k) \) can be rewritten as

\[ y(k) = \sum_{i=1}^{c} \omega_{i}(x(k)) x(k) \Theta_i, \]

where for \( i = 1, \ldots, c \), and assuming Gaussian membership functions,

\[ x(k) = [x_1(k), \ldots, x_N(k)]^T, \]

\[ \mu_{A_{ij}}(x_j) = \exp \left( \frac{-(x_j - v_{ij})^2}{\sigma_{ij}} \right), \quad j = 1, \ldots, N, \]

\[ \omega_{i}(x(k)) = \frac{\prod_{j=1}^{N} \mu_{A_{ij}}(x_j)}{\sum_{i=1}^{c} \prod_{j=1}^{N} \mu_{A_{ij}}(x_j)}, \]

\[ \Theta_i = [\theta_{i1}, \ldots, \theta_{iN}]^T, \]

\[ \Psi(k) = \{\omega_{i}(x(k)) x(k), \ldots, (\omega_{c}(x(k)) x(k)\} \]

where \( v_{ij} \) and \( \sigma_{ij} \) are the antecedent parameters, which represent the center and width of the antecedent membership functions, respectively, and which need to be defined/learned. Parameters \( v_{ij} \) and \( \sigma_{ij} \) will be learned by the NUFC method which uses a K-nearest neighbor and the K-Fuzzy C-means method presented in Section 2.2 and Section 2.3, respectively.

2.2. Fuzzy C-Means

The objective of the fuzzy c-means (FCM) clustering algorithm is the partitioning of a dataset \( X \) into a predefined number of clusters, \( c \). In the fuzzy clustering methods, the objects can belong to multiple clusters, with different degrees of membership. Consider \( N \) values which compose an observation/object/sample \( l \) (one value for each input variable), which are grouped into an N-dimensional observation/sample vector \( x_l = [x_{l1}, \ldots, x_{lN}]^T \in \mathbb{R}^N \). A set of \( L \) observations/objects is then denoted as

\[ X = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1N} \\ x_{21} & x_{22} & \ldots & x_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{L1} & x_{L2} & \ldots & x_{LN} \end{bmatrix}. \]

A fuzzy partition of the set \( X \) into \( c \) clusters, is a family of fuzzy subsets \( \{A^i| 1 \leq i \leq c \} \). The membership functions of these fuzzy subsets are defined as \( \mu_i(l) = \mu_{A^i}(x_l) \), and form the fuzzy partition matrix \( U = [u_{il}] = [\mu_i(l)] \in \mathbb{R}^{c \times L} \). The \( i \)-th row of the matrix \( U \) contains the values of the membership function of the \( i \)-th fuzzy subset \( A^i \) for all the observations belonging to the data matrix \( X \).

The partition matrix has to meet the following conditions (Dovžan & Škrjanc, 2011): The membership degrees are real numbers in the interval \( \mu_i(l) \in [0, 1] \), for \( l = 1, \ldots, L \); The total membership of each sample in all the clusters must be equal to one \( \sum_{i=1}^{c} \mu_i(l) = 1 \); And none of the fuzzy clusters is empty, neither do any contain all the data \( 0 < \sum_{l=1}^{L} \mu_i(l) < L \), for \( i = 1, \ldots, c \). FCM clustering tries to minimize the following objective function, which has a pre-defined number of clusters, \( c \), and includes a fuzziness parameter, \( \eta \):

\[ J(X, U, V) = \sum_{i=1}^{c} \sum_{l=1}^{L} [\mu_i(l)]^\eta d_{il}^2(x_l, v_i), \]

where \( V = [v_{11}, \ldots, v_{c}]^T \in \mathbb{R}^{c \times n} \) is a matrix of cluster centroid vectors \( v_i = [v_{i1}, \ldots, v_{in}]^T \), \( d_{il}(x_l, v_i) \) is the Euclidean distance (\( l^2 \)-norm) between the observation \( x_l \) and the cluster centroid \( v_i \), and the overlapping factor or the fuzziness parameter \( \eta \) influences the fuzziness of the resulting partition. The partition can range from a hard partition (\( \eta = 1 \)) to a completely fuzzy partition (\( \eta \rightarrow \infty \)). In order to find the fuzzy clusters in the dataset \( X \), equation (10) must be minimized. If the derivative of objective function is taken with respect to the cluster centers \( V \) and to the membership values \( U \), then optimum membership values are calculated as follows (Dovžan & Škrjanc, 2011):

\[ \mu_i(l) = \left( \sum_{q=1}^{c} (d_{iq}^2)^{(1/(\eta-1))} \right)^{-1}, \]
where

\[ d_{ij}^2 = (x_i - v_j)^T (x_i - v_j), \]

and

\[ v_i = \frac{\sum_{l=1}^L \mu_i^2(l) x_i}{\sum_{l=1}^L \mu_i^2(l)}. \]

The \( v_{ij} \) parameters of (4) are obtained from the center-vectors \( v_i = [v_{i1}, \ldots, v_{iN}]^T \) of (13). To finalize the identification of the premise parameters in (4) of the T-S model characteristics regarding all of the data, internal criteria can be used to evaluate the quality of the clustering that results from the algorithm, some cluster validity index is required to evaluate a clustering algorithm by comparing it to other clustering schemes. However, all of these criteria can be converted into two main optimal clustering criteria, compactness and separation (Berry & Linoff, 1996). Compactness criteria has been proposed based on the idea that the members in one same cluster should be as close to each other as possible. For the compactness criteria the variance, which should be minimized, is one common measure. Some conventional fuzzy validity index for this class of scheme are for example the partition coefficient (PC) (Bezdek, 1974), the partition entropy (Bezdek, 1975), and the proportion exponent (Windham, 1982). Separation criteria are organized based on the distance between the closest members of the clusters, the distance between the most distant members, or the distance between the centers of the clusters. A conventional performance index in this class of scheme is the Xie-Beni index (Xie & Beni, 1991). Although the Xie-Beni index has proved that it can provide reliable response over a wide range of choices for the number of clusters and fuzziness weighting exponent, the Xie-Beni index has two intrinsic drawbacks: 1) the validation index monotonically decreases when the number of clusters gets very large and close to the number of data points, and 2) strong interaction between the cluster validity index and fuzziness weighting exponent \( \eta \) imposes unpredictable behaviour in results when fuzziness weighting exponent approaches infinity. The first problem was considered by Kwon (1998), who imposed a punishing function to eliminate the decreasing tendency. Tang & Sun (2005) proposed an improved validation index for the FCM algorithm to overcome the above two problems with the same idea. This work uses the Tang-Sun’s validation index which is defined as follows:

\[
V_T(U, V; X) = \frac{1}{c} \sum_{i=1}^c \sum_{k=1}^c \mu_{ik}^2 \| x_i - v_k \|^2 + 1/c \min_{i \neq k} \| v_i - v_k \|^2 + 1/c.
\]

where \( c \) is number of clusters, and \( v_k \) is the center of cluster \( k \). The numerator of the second term in (15) is an ad hoc punishing function (average distance between cluster centers) which is applied to eliminate the decreasing tendency of \( V_T(\cdot) \) as \( c \to L \).

2.4. K-Nearest Neighbor (K-NN)

The FCM algorithm with pre-defined initial values such as the number of clusters, initial cluster centers and fuzziness weighting exponent \( \eta \) converges to a solution at which the objective function \( J \) in (10) is minimized. In practice, in many cases, randomly choosing initial FCM parameters may cause the FCM to just obtain results which are only locally optimal (Yu et al., 2004; Wu, 2012). To overcome these problems, an initialization technique based on a K-NN method is proposed to initialize the FCM method [Algorithm 1, Steps 1-3c]. The basic FCM follows an iterative procedure to converge to a solution. But with the proposed initialization procedure just one iteration of the FCM is enough to learn the antecedent parameters of the in the course of learning the T-S fuzzy model [Algorithm 1, Step 3d]. The basic K-NN is a non parametric learning algorithm for classification with random initialization of the ‘K’ which decides how many neighbours influence the classification. While, with the proposed initialization procedure the ‘K’ is a desirable parameter which will be iteratively calculated by the proposed method with the goal of obtaining the best/optimal value, \( c^* \), for the number of clusters \( c \) subject to a maximum \( c_{\text{max}} \) (see Algorithm 1, Steps 3). The resulting \( K \) for the best \( c \) is obtained in Algorithm 1. The complete proposed method is explained in the Section 4. In Section 4 it will be explained how the combination of these steps together can result the NUFCA to initialize the T-S fuzzy modelling.

3. Recursive Least Squares Method With Adaptive Directional Forgetting

In the proposed nonlinear systems modeling methodology, after learning the antecedent parameters, the consequent parameters are given by a recursive least squares (RLS) method, with the adaptive directional forgetting approach of (Kulhavý, 1987; Bobál et al., 2005) here adopted
for the T-S fuzzy model. Using off-line training algorithms, the T-S fuzzy model can be obtained from input-output data collected from a plant. However, such collected dataset(s) can be limited, the obtained T-S fuzzy models may not provide adequate accuracy, the system can be nonlinear and/or time-varying, and can have varying operating points and varying model parameters. Adaptive methodologies should be applied to solve these problems. At each iteration $l$, the vector of parameter estimations (6), is updated using

$$\theta_i(l) = \theta_i(l - 1) + \frac{C_i(l - 1)\psi_i^T(l)}{1 + \xi_i}[y_i(l) - \psi_i(l)\theta_i(l - 1)],$$

where $\psi_i(l) = (\omega^i(x(l)))x(l)$, $\xi_i = \psi_i(l)C_i(l - 1)\psi_i^T(l)$, $C_i(l)$ is the covariance matrix of fuzzy rule $i$, and $y_i(l) = (\omega^i(x(l)))y(l)$.

The covariance matrix is also updated at each iteration, $l$, using

$$C_i(l) = C_i(l - 1) - \frac{C_i(l - 1)\psi_i^T(l)\psi_i(l)C_i(l - 1)}{\varepsilon_i + \xi_i},$$

where $\varepsilon_i = \varphi_i(l - 1) - \frac{1}{\xi_i}$, and $\varphi_i(l - 1)$ is the forgetting factor at iteration $(l - 1)$ of the fuzzy rule $i$.

The adaptation performed on the forgetting factor is obtained using (Kulhavý, 1987; Bobál et al., 2005):

$$\varphi_i(l) = \frac{1}{1 + (1 + \rho) \left\{ \ln(1 + \xi_i) + \frac{(\nu_i(l) + 1)\gamma_i}{1 + \xi_i} - 1 \right\}},$$

where $\nu_i(l) = \varphi_i(l - 1)\gamma_i(l - 1) + 1$, $\gamma_i = \frac{y_i(l) - \psi_i(l)\theta_i(l - 1)}{\tau_i(l)}, \tau_i(l) = \varphi_i(l - 1) + \frac{y_i(l) - \psi_i(l)\theta_i(l - 1)}{1 + \xi_i}$, and $\rho$ is a positive constant.

4. Proposed T-S Fuzzy Model Identification Algorithm

In this section a new T-S fuzzy model identification algorithm is proposed. To construct a T-S fuzzy system of the form (2), it is necessary to obtain the number of rules, the antecedent membership functions, the set of rules, and also to update the consequent parameters ($\theta_i$). The antecedent part is given by a new unsupervised fuzzy clustering algorithm (NUFCA) and the consequent parameters are estimated by the recursive least squares method with adaptive directional forgetting (Sec. 3). The complete proposed algorithm for T-S fuzzy model identification is presented in Algorithm 1. The NUFCA uses a hybrid clustering algorithm based on two layers (Algorithm 1, Steps 1-4). NUFCA iteratively tests several values for the number of clusters $c$, in order to find an optimal value which is denoted as $c^\ast$. In the first layer of NUFCA, for each $c$, the initial centers of the clusters are obtained by using a technique based on the KNN approach (Step 3b). The basic idea here is to try to find, from $L$ samples of a dataset, the $k$ samples which have the highest levels of similarity to a specified feature vector. Specifically, in the

Algorithm 1 Proposed T-S fuzzy model identification Algorithm.

1. Construct the matrix $X = [x_l]_{l \times N}$, $1 \leq l \leq L$ and $1 \leq j \leq N$, in (9) using $L$ observations;
2. Choose the degree of fuzziness $\gamma > 1$; And let $g_{i0}$ be the center of data $X$, and $v_{ij} \leftarrow 0$;
3. Repeat the procedure below for $c = 1, 2, \ldots, c_{\max} = \sqrt{L}$:
   (a) Initialization for iteration $c$:
      i. Let $K = \frac{\ell}{2} - 1$, and $I = \{1, 2, \ldots, L\}$, where $[\cdot]$ is the floor function;
   (b) For $i = 1, \ldots, c$ construct $E_i$ using $K$ nearest neighbourhood:
      i. In $I$ find the index $i$ of the unknown sample $x^\prime$ which is farthest from $g_{0i}$;
      ii. $E_i = \{x^\prime\} \cup \text{KNN}(K - 1, x^\prime)$, where $\text{KNN}(K - 1, x^\prime)$ is the set of $K - 1$ nearest-neighbour samples of $x^\prime$ that do not belong to any other already existing $E_i$;
      iii. Let $g_{i0} = \sum_{x_l \in E_i} x_l / K$, and $E_i \leftarrow E_i \cup \{g_{i0}\}$;
      iv. $I \leftarrow I \setminus \{i\} \setminus \text{KNN}(K - 1, x^\prime)$, where $\text{KNN}(K - 1, x^\prime)$ is the set of all indices $n$ such that $x^n \in \text{KNN}(K - 1, x^\prime)$;
   (c) While $I \neq \emptyset$, do:
      i. Select $r \in I$, let $I \leftarrow I \setminus \{r\}$, and calculate the distances from the still unclustered sample $x^\prime$ to the center $g_{ir}$ of all $E_i$, by $d(x^\prime, g_{ir})$, $\forall i = 1, \ldots, c$;
      ii. Assign $x^\prime$ to the $E_i$ with the nearest $g_{ir}$, so that $E_i \leftarrow E_i \cup \{x^\prime\}$;
      iii. Perform the update of $g_{ir} = \sum_{x_l \in E_i} x_l / K + 1$;
   (d) Perform one iteration of FCM:
      i. Calculate the fuzzy clustering matrix $U = [u_{il}]_{c \times L}$ using (11)-(12) with $v_{il} = g_{il}$ in (12);
      ii. Calculate clustering validity index by (15) and assign it to $v_{il}$;
   (e) If $v_{il} > v_{i\ast}^{\prime}$, then
      i. Let the optimal number of clusters be $c^\ast \leftarrow c^\ast$;
      ii. Let $E_{i^\ast}^\prime \leftarrow E_i$, for $i = 1, \ldots, c^\ast$, be the optimal clustering sets;
      iii. Update the optimal clustering validity index: $v_{i^\ast}^\prime \leftarrow v_{i^\ast}
4. Using $U = [u_{il}]_{c^\ast \times L}$ calculate $v_{i}$ and $\sigma_{ij}$ by (13)-(14).
5. Compute the consequent parameters $\theta_i$, by initializing its components to small values, and then using the recursive least squares method with adaptive directional forgetting (Section 3), using recursion (16) for $l = 1, \ldots, L$;
first layer of NUFCA, dataset \( X \) of (9) is partitioned into \( c \) clusters, in which samples of each cluster have similarity in the Euclidean distance sense, and will belong to one set \( E_i \) (Steps 3a-3c). \( E_i \) is an auxiliary set of samples to gather the members of tentative cluster \( i \). After all \( E_i \) sets are constructed for a certain \( c \), then one iteration of FCM is performed (Step 3d). The final step of NUFCA consists on determining the best \( c \) and the corresponding collection of the best \( E_i \) (\( i = 1, \ldots, c \)), which are termed as \( c^* \) and \( E_i^* \) (\( i = 1, \ldots, c^* \)), respectively (Step 3e). The results of this proposed hybrid clustering algorithm are used to set the antecedent parameters of the T-S model (1)-(2). In the final step, the algorithm uses the RLS procedure to obtain consequent parameters of T-S fuzzy model (Step 5).

5. Adaptive Fuzzy Predictive Control Law

After having studied the identification algorithm (Sections 2-4), in this section the control algorithm is presented. A diagram of the adaptive fuzzy generalized predictive control (AFGPC) approach is presented in Figure 1. As can be seen, the control scheme consists of the plant, the controller, and the adaptive T-S fuzzy model. The controller is composed of a model-based predictive controller that integrates a T-S fuzzy model learned off-line, according to the methodology presented on Algorithm 1. Also in online mode, the consequent parameters of the model are adjusted in a recursive procedure using the adaptation law studied in Section 3. The main steps of the control architecture are presented at the end of this section, in Algorithm 2.

A large class of nonlinear processes can be represented by a model of the following type:

\[
y(k) = f[y(k - 1), y(k - 2), \ldots, y(k - n_y), u(k - d - 1), \ldots, u(k - d - n_u)],
\]

where \( u(\cdot) : \mathbb{N} \to \mathbb{R} \) and \( y(\cdot) : \mathbb{N} \to \mathbb{R} \) are the process input and output, respectively, \( n_y \in \mathbb{N} \) and \( n_u \in \mathbb{N} \) are the orders of the input and output, respectively, and \( d \in \mathbb{N} \) is the time-delay of the system. In this discrete-time nonlinear SISO plant (19), \( f(\cdot) : \mathbb{R}^{n_y+n_u} \to \mathbb{R} \) represents a nonlinear mapping which is assumed to be unknown. \( f(\cdot) \) is approximated by a T-S fuzzy system. For the GPC controller, system (19) can be described by a T-S fuzzy model defined by the following fuzzy rules:

\[
R_i : \text{IF } x_i(k) \text{ is } A_{i1}, \ldots \text{ and } x_N(k) \text{ is } A_{iN} \text{ THEN } y_i(k) = a_i(z^{-1})y(k - 1) + b_i(z^{-1})u(k - d - 1), \quad i = 1, \ldots, c
\]

where \( c \) is the number of rules, \( N = n_y + n_u \),

\[
a_i(z^{-1}) = a_{i1} + a_{i2}z^{-1} + \ldots + a_{i n_y}z^{-(n_y - 1)},
\]

\[
b_i(z^{-1}) = b_{i1} + b_{i2}z^{-1} + \ldots + b_{i n_u}z^{-(n_u - 1)},
\]

and \( x(k) = [x_1(k), \ldots, x_N(k)] = [y(k - 1), \ldots, y(k - n_y), u(k - d - 1), \ldots, u(k - d - n_u)] \) is the vector of input variables of the T-S fuzzy system. Thus, from (20) \( y(k) \) can be rewritten as

\[
y(k) = \sum_{i=1}^{c} \hat{\omega}_i(x(k)) [a_i(z^{-1})y(k - 1) + b_i(z^{-1})u(k - d - 1)],
\]

\[
= \sum_{i=1}^{c} \hat{\omega}_i(x(k)) \Psi(k) \Theta_i,
\]

\[
\Psi(k) = [\hat{\omega}_1(x(k)), \hat{\omega}_2(x(k)), \ldots , \hat{\omega}_c(x(k))].
\]

where for \( i = 1, \ldots, c \),

\[
\hat{\omega}_i(x(k)) = \frac{\prod_{j=1}^{N} A_{ij}(x)}{\sum_{i=1}^{c} \prod_{j=1}^{N} A_{ij}(x)},
\]

\[
\Theta_i = [a_{i1}, \ldots, a_{i n_y}, b_{i1}, \ldots, b_{i n_u}]^T,
\]

\[
\Psi(k) = [\Theta_1(k), \Theta_2(k), \ldots , \Theta_c(k)]^T,
\]

\[
\hat{\omega}_i(x(k)) = \prod_{j=1}^{N} \lambda_j \Delta u(k + p - d - 1 | k)^2,
\]

5.1. Predictive Control Law

It is assumed that the plant model is of the form (23), which can be rewritten as follows:

\[
\bar{a}(z^{-1})y(k) = \bar{b}(z^{-1})u(k - d - 1),
\]

where

\[
\bar{a}(z^{-1}) = 1 - a_{11}z^{-1} - \ldots - a_{n_y}z^{-n_y},
\]

\[
\bar{b}(z^{-1}) = b_1 + b_2z^{-1} + \ldots + b_{n_u}z^{-(n_u - 1)},
\]

\[
\bar{a}_i = \sum_{j=1}^{c} \hat{\omega}_i(x(k)) a_{1j}, \quad t = 1, \ldots, n_y,
\]

\[
\bar{b}_m = \sum_{i=1}^{c} \hat{\omega}_i(x(k)) b_{mj}, \quad m = 1, \ldots, n_u.
\]

The GPC control law is obtained so as to minimize the following cost function

\[
J(k) = \sum_{p=d+1}^{d+ Ng} [\hat{y}(k + p | k) - r(k + p)]^2 + \lambda (z^{-1}) \Delta u(k + p - d - 1 | k)^2
\]

where \( \hat{y}(k + p | k) \) is an \( p \)-step ahead prediction of the system on instant \( k \), \( r(k + p) \) is the future reference trajectory, \( \Delta = 1 - z^{-1} \) and \( \lambda(z^{-1}) = \lambda_0 + \lambda_1 z^{-1} + \ldots + \lambda_{N_y+n_u-1} z^{-(N_y+n_u-1)} \) is a weighting polynomial. \( N_p \) and \( N_y \) are the output and control horizons, respectively. Consider the following Diophantine equation (34):

\[
1 = \Delta e_p(z^{-1}) \bar{a}(z^{-1}) + z^{-p} \bar{f}(z^{-1}),
\]

\[
e_p(z^{-1}) = 1 + e_{11}z^{-1} + \ldots + e_{p-1}z^{-(p-1)},
\]

\[
f_p(z^{-1}) = f_{p,0} + f_{p,1}z^{-1} + \ldots + f_{p,n_u}z^{-n_u},
\]
where $e_p(z^{-1})$ and $f_p(z^{-1})$ can be obtained by dividing 1 by $\Delta \hat{a}(z^{-1})$ until the remainder can be factorized as $z^{-p}f_p(z^{-1})$. The quotient of the division is the polynomial $e_p(z^{-1})$. A simple and efficient way to obtain polynomials $e_p(z^{-1})$ and $f_p(z^{-1})$ is to use recursion of the Diophantine equation as demonstrated in (Camacho & Bordons, 1998). Polynomials $e_p(z^{-1})$ and $f_p(z^{-1})$ can be obtained from polynomials of $e_p(z^{-1})$ and $f_p(z^{-1})$, respectively. Polynomials $e_{p+1}(z^{-1})$ are given by

$$e_{p+1}(z^{-1}) = e_p(z^{-1}) + z^{-p}e_{p+1,p},$$

where $e_{p+1,p} = f_{p,0}$. The coefficients of polynomial $f_{p+1}(z^{-1})$ can be obtained recursively as follows:

$$f_{p+1,i} = f_{p,i+1} - f_{p,0} \Delta \bar{a}_{i+1}, \quad i = 0, \ldots, n_y - 1,$$

where $f_{p,n_y} = 0$. Polynomial $g_{p+1}(z^{-1})$ is expressed as:

$$g_{p+1}(z^{-1}) = e_{p+1}(z^{-1})b(z^{-1}),$$

$$= \left[e_p(z^{-1}) + z^{-p}f_{p,0}\right]b(z^{-1}),$$

$$= g_p(z^{-1}) + z^{-p}f_{p,0}b(z^{-1}),$$

where the coefficients of $g_{p+1}(z^{-1})$ are given by $g_{p+1,j} = g_{p,j}$ for $j = 0, \ldots, p - 1$, and

$$g_{p+1,p+i} = g_{p,p+i} + f_{p,0}b, \quad i = 0, \ldots, n_u,$$

where $g_{p,p+n_u} = 0$. $e_p(z^{-1})$, $f_p(z^{-1})$, and $g_p(z^{-1})$ are recursively computed for $p = d + 1, \ldots, N_p$. To initialize the recursion (34), $p = d + 1$, and

$$e_{d+1}(z^{-1}) = 1,$$

$$f_{d+1}(z^{-1}) = z(1 - \tilde{a}(z^{-1})),$$

$$= \tilde{a}_1 + \tilde{a}_2z^{-1} + \ldots + \tilde{a}_n z^{-n},$$

where

$$\tilde{a}(z^{-1}) = \Delta \hat{a}(z^{-1}) = 1 - \tilde{a}_1z^{-1} - \ldots - \tilde{a}_n z^{-n+1}.$$

Thus,

$$g_{d+1}(z^{-1}) = e_{d+1}(z^{-1})\tilde{b}(z^{-1}) = \tilde{b}(z^{-1}).$$

Multiplying (28) by $\Delta z^p e_p(z^{-1})$ yields

$$\Delta z^p e_p(z^{-1})\tilde{a}(z^{-1}) y(k) = \Delta z^p e_p(z^{-1})\tilde{b}(z^{-1}) u(k - d - 1).$$

Defining

$$g_p(z^{-1}) = e_p(z^{-1})\tilde{b}(z^{-1}),$$

$$= g_{p,0} + g_{p,1}z^{-1} + \ldots + g_{p,p+n_u} z^{-p+n_u-1},$$

and substituting (34) and (47) into (46) gives

$$y(k+p|k) = f_p(z^{-1})y(k) + g_p(z^{-1})\Delta u(k + p - d - 1).$$

Thus, the best prediction of $y(k+p|k)$ is

$$\hat{y}(k+p|k) = f_p(z^{-1})y(k) + g_p(z^{-1})\Delta u(k + p - d - 1).$$

Equation (50) can be rewritten as

$$y(k) = Gu(k) + F(z^{-1})y(k) + L(z^{-1}).$$
where

\[
y(k) = \begin{bmatrix}
y(k + d + 1) \\
y(k + d + 2) \\
\vdots \\
y(k + N_p)
\end{bmatrix},
\]

(52)

\[
u(k) = \begin{bmatrix}
\Delta u(k) \\
\Delta u(k + 1) \\
\vdots \\
\Delta u(k + N_u - 1)
\end{bmatrix},
\]

(53)

\[
G = \begin{bmatrix}
g_{1,0} & 0 & \cdots & 0 \\
g_{2,1} & g_{2,0} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
g_{N_p,N_p-1} & g_{N_p,N_p-2} & \cdots & g_{N_p,N_p-N_u}
\end{bmatrix},
\]

(54)

\[
L = \begin{bmatrix}
[g_{d+1}(z^{-1}) - \hat{g}_{d+1}(z^{-1})] \Delta u(k-1) \\
[g_{d+2}(z^{-1}) - \hat{g}_{d+2}(z^{-1})] \Delta u(k-1) \\
\vdots \\
[g_{N_p}(z^{-1}) - \hat{g}_{N_p}(z^{-1})] \Delta u(k-1)
\end{bmatrix},
\]

(59)

\[
\hat{g}_p(z^{-1}) = g_{p,0} + g_{p,1}z^{-1} + \ldots + g_{p,N_p-1}z^{d+1-p}.
\]

Using (51) and considering \( \lambda(z^{-1}) \) to be constant (\( \lambda > 0 \)), (33) can be rewritten as

\[
J_{eq}(k) = [Fy(k) + Gu(k) + L - R]^T [Fy(k) + Gu(k) + L - R] + [\lambda u(k)]^2,
\]

(55)

where

\[
R = [r(k + d + 1), \ldots, r(k + N_p)]^T.
\]

(56)

To minimize \( J_{eq}(k) \) the following equation is solved

\[
\frac{\partial J_{eq}(k)}{\partial \Delta u(k)} = 0.
\]

(57)

By minimizing \( J_{eq}(k) \) using (57), the following optimum control increment is obtained (Camacho & Bordons, 1998):

\[
u^*(k) = \frac{G^T (R - Fy(k) - L) - L}{G^T G + \lambda I},
\]

(58)

where \( I \) is the identity matrix. As the control signal sent to the process is the first row of \( u^*(k) \), the \( \Delta u^*(k) \) is given by:

\[
\Delta u^*(k) = K [R - Fy(k) - L],
\]

(59)

where \( K \) is the first row of matrix \((G^T G + \lambda I)^{-1}G^T\).

\[
K = [1 \ 0 \ \cdots \ 0]_{1 \times N_u} (G^T G + \lambda I)^{-1}G^T.
\]

(60)

Algorithm 2 summarizes the design and operation of the adaptive fuzzy generalized predictive control method.

---

**Algorithm 2** Adaptive fuzzy generalized predictive control algorithm.

1. Design control parameters: \( N_p, N_u, \lambda \) and \( d \). Design the identification parameters \((\rho, \varphi_i, \tau, \nu_i, \text{ for } i = 1, \ldots, c)\) with the same values as the ones defined in Algorithm 1;

2. Use the fuzzy rule base (input variables, respectively membership functions, the fuzzy rules and the final learned model parameters) learned in Algorithm 1 and initialize \( u(0) \);

3. For each newly arriving online sample, do:

(a) Compute \( \hat{a}(z^{-1}) \) and \( \hat{b}(z^{-1}) \) using (29) and (30), respectively;

(b) Compute control signal \( \Delta u(k) \) with (59);

(c) Adapt the T-S fuzzy model parameters \((a_{ji} \text{ and } b_{ji} \text{ of (21)})\) by performing one iteration of recursion (16).

---

**6. Experiments and Results**

This section presents simulation and real-world results to demonstrate the feasibility, performance, and effectiveness of the proposed T-S design methodology in identification and in control. First, a nonlinear system identification application problem is analysed and quantitatively compared with two adaptive approaches: a recurrent partial least squares (RPLS) (Dayal & MacGregor, 1997), and a recently proposed incremental local learning soft sensing algorithm (ILLSA) for adaptive soft sensors (Kadlec & Gabrys, 2011). Furthermore, the results with two other methods: a new fuzzy c-regression model algorithm (NFCRMFA) (Li et al., 2009), and a hierarchical genetic approach (MGA) for learning T-S fuzzy models was proposed by (Mendes et al., 2012) are compared in the estimation of the floor concentration in the effluent of a real-world wastewater treatment plant. Then, the performance of the proposed adaptive predictive fuzzy identification and control methodology is demonstrated on two setups: a simulated CSTR plant, and a real-world experimental setup composed of two coupled DC motors. For both applications, the proposed identification performance is quantitatively compared with the aforementioned methods. To better demonstrate the efficiency of the NFCA, also results of experiments regarding the combination of the NFCA with other methods are presented. Moreover, next to presentation of the proposed AFGPC, a comparison between performance of the AFGPC and the classical GPC (Camacho & Bordons, 1998) is exhibited and discussed.
The selected degree of fuzziness was set to $16/12/800/4/18/10/600/200/6$.

Fitness evaluation of the WWTP against the number of fitness function $(1/L) \sum_{k=1}^{L} (y_k - \hat{y}_k)^2$, in the test dataset is obtained with the method proposed in this paper, where $y_k$ and $\hat{y}_k$ are the real and predicted values of $y$ at instant $k$, respectively. Also, comparing with the HGA, the proposed method uses a lower number of variables and fuzzy rules, but shows a better performance.

Another evaluation is considered in Figure 3 which illustrates trajectory of fitness evaluation in the WWTP when number of clusters considered in the input WWTP data was changed. In fact the first purpose to design of the NUFCA was to give address to concerns with the T-S fuzzy modelling based on FCM. However, the result in Figure 3 shows that finding the optimal cluster number can improve estimation of the WWTP plant. This parameter plays an important role in the design of the fuzzy membership functions, and individual rules. As it can be seen in range from the $c = 2$ until $c = 11$ fitness evaluation values are improving. After $c = 11$ the optimal

<table>
<thead>
<tr>
<th>Variables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>Amount of chlorine in the influent;</td>
</tr>
<tr>
<td>$u_2$</td>
<td>Amount of chlorine in the effluent;</td>
</tr>
<tr>
<td>$u_3$</td>
<td>Turbidity in the raw water;</td>
</tr>
<tr>
<td>$u_4$</td>
<td>Turbidity in the influent;</td>
</tr>
<tr>
<td>$u_5$</td>
<td>Turbidity in the effluent;</td>
</tr>
<tr>
<td>$u_6$</td>
<td>Ph in the raw water;</td>
</tr>
<tr>
<td>$u_7$</td>
<td>Ph in the influent;</td>
</tr>
<tr>
<td>$u_8$</td>
<td>Ph in the effluent;</td>
</tr>
<tr>
<td>$u_9$</td>
<td>Color in the raw water;</td>
</tr>
<tr>
<td>$u_{10}$</td>
<td>Color in the influent;</td>
</tr>
<tr>
<td>$u_{11}$</td>
<td>Color in the effluent;</td>
</tr>
<tr>
<td>$y$</td>
<td>Flour in the effluent.</td>
</tr>
</tbody>
</table>

6.1. Application to Wastewater Treatment System

In this section, the performance of the proposed identification methodology is studied. Specifically, a Soft Sensor application is studied. The objective of this experiment is to estimate the flour concentration in the effluent of a real-world urban wastewater treatment plant (WWTP).

The dataset of plant variables that is available for learning consists of 11 input variables, $u_1, \ldots, u_{11}$, and one target output variable to be estimated, $y$. The variables correspond to physical values, such as pH, turbidity, color of the water and others. The input variables are measured on-line by plant sensors, and the output variable in the dataset is measured by laboratory analysis. The plant variables are described in Table 1. The historical data set comprises three years of acquisition, with a sample rate of 2 hours for the variables acquired by sensors (input variables). The target variable, the flourine, is laboratory measured at every 24 hours. Complete information about the WWTP can be found in (Souza et al., 2013; Souza & Araújo, 2014).

To construct the dataset, the first three delayed versions of each variable were chosen as candidates for inputs of the T-S model. Specifically, the following combinations of process variables and delays are used as the candidates for inputs of the T-S model to predict $y(t)$: $[u_1(t-1), u_1(t-2), u_1(t-3), \ldots, u_{11}(t-1), u_{11}(t-2), u_{11}(t-3)]$. The available data set was split into 30% of data for training, and the remaining 70% of data was used to test the proposed algorithm. The selected degree of fuzziness was set to $\eta = 2$, and the optimal number of clusters that resulted from Algorithm 1 was $c = 13$. Figure 2 shows the predicted and desired (real) values of the target variable to be estimated, for the WWTP experiment. As can be seen, the accuracy of the modeling is acceptable. Numerical results comparing the performance of the proposed method and the works RPLS (Dayal & MacGregor, 1997), NFCRMA (Li et al., 2009), ILLSA (Kadlec & Gabrys, 2011), and HGA (Mendes et al., 2012) are presented in Table 2.

As can be seen comparing with other methods, the largest value of the fitness function $(1/MSE)$, $MSE = (1/L) \sum_{k=1}^{L} (y_k - \hat{y}_k)^2$, in the test dataset is obtained with the method proposed in this paper, where $y_k$ and $\hat{y}_k$ are the real and predicted values of $y$ at instant $k$, respectively. Also, comparing with the HGA, the proposed method uses a lower number of variables and fuzzy rules, but shows a better performance. Another evaluation is considered in Figure 3 which illustrates trajectory of fitness evaluation in the WWTP when number of clusters considered in the input WWTP data was changed. In fact the first purpose to design of the NUFCA was to give address to concerns with the T-S fuzzy modelling based on FCM. However, the result in Figure 3 shows that finding the optimal cluster number can improve estimation of the WWTP plant. This parameter plays an important role in the design of the fuzzy membership functions, and individual rules. As it can be seen in range from the $c = 2$ until $c = 11$ fitness evaluation values are improving. After $c = 11$ the optimal
trajectory evolves with small changes. However, for this case study, the best result was obtained for \( c^* = 13 \).

6.2. Control of a Continuous-Stirred Tank Reactor (CSTR)

A Continuous Stirred Tank Reactor (CSTR) is a highly nonlinear process which is very common in chemical and petrochemical plants. In the process, a single irreversible, exothermic reaction is assumed to occur in the reactor. The CSTR for an exothermic irreversible reaction \( A \rightarrow B \) is described by the following dynamic model based on a component balance for reactant \( A \) and on an energy balance (Morningred et al., 1992):

\[
\frac{\partial C_A(t + d_c)}{\partial t} = \frac{q(t)}{V} (C_{A0} - C_A(t + d_c)) - k_0 C_A(t + d_c) \exp \left(-\frac{E}{RT(t)}\right) + \vartheta(t),
\]

\[
\frac{\partial T}{\partial t} = \frac{q(t)}{V} (T_0(t) - T(t)) - \frac{(-\Delta H) k_0 C_A(t + d_c)}{\rho c_p} \exp \left(-\frac{E}{RT(t)}\right) \left[1 - \frac{\vartheta(t)}{c_p(t)\rho c_p} \right] + \frac{\vartheta(t)}{\rho c_p} \left[T_{ci}(t) - T(t)\right],
\]

\[
y(t) = C_A(t), \quad u(t) = q_c(t).
\]  

A stochastic disturbance \( \vartheta(t) \), namely a Gaussian white noise, was also considered in (61). The objective is to control the measured concentration of \( C_A(t) \) by manipulating the coolant flow rate \( q_c(t) \). The plant variables and the respective nominal values are described in Table 3. The sampling period is assumed to be \( T = 0.1 \text{ min} \), and the time delay is assumed to be \( d_c = 5T = 0.5 \text{ min} \).

6.2.1. Identification

As a first step, a dataset representative of the CSTR operation was constructed. The dataset was obtained by applying the control signal represented in Figure 4a.

Table 2: Comparison results on the test dataset for the wastewater treatment plant (WWTP).

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of rules</th>
<th>Number of inputs</th>
<th>Inputs</th>
<th>1/MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPLS (Dayal &amp; MacGregor, 1997)</td>
<td>-</td>
<td>-</td>
<td>all candidate variables</td>
<td>840.9</td>
</tr>
<tr>
<td>NFCRMA (Li et al., 2009)</td>
<td>-</td>
<td>-</td>
<td>all candidate variables</td>
<td>206.13</td>
</tr>
<tr>
<td>ILLSA (Kadlec &amp; Gabrys, 2011)</td>
<td>-</td>
<td>-</td>
<td>all candidate variables</td>
<td>1197.6</td>
</tr>
<tr>
<td>HGA (Mendes et al., 2012)</td>
<td>20</td>
<td>13</td>
<td>( c(t-1), u_4(t-2), u_4(t-3), u_6(t-3), )( \vartheta(t-1), \vartheta(t-3), \vartheta(t-3), )( u_9(t-1), u_9(t-2), u_{10}(t-2), u_{11}(t-3), u_{11}(t-2) )</td>
<td>901.2</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>13</td>
<td>11</td>
<td>( u_1(t-1), u_1(t-3), u_3(t-1), u_6(t-3), )( \vartheta(t-1), \vartheta(t-3), u_6(t-1), u_6(t-3), )( u_9(t-3), u_{10}(t-1), u_{11}(t-3) )</td>
<td>1716.2</td>
</tr>
</tbody>
</table>

Table 3: Variables of the continuous stirred tank reactor (CSTR) (Morningred et al., 1992).

<table>
<thead>
<tr>
<th>Variables-Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_A ) - Product concentration</td>
<td>0.1 [mol/l]</td>
</tr>
<tr>
<td>( T ) - Reactor temperature</td>
<td>438.54 [K]</td>
</tr>
<tr>
<td>( q_c ) - Coolant flow rate</td>
<td>1034.41 [l/min]</td>
</tr>
<tr>
<td>( q ) - Process flow rate</td>
<td>100 [l/min]</td>
</tr>
<tr>
<td>( C_{A0} ) - Feed concentration</td>
<td>1 [mol/l]</td>
</tr>
<tr>
<td>( T_{io} ) - Feed temperature</td>
<td>350 [K]</td>
</tr>
<tr>
<td>( T_{io} ) - Inlet coolant temperature</td>
<td>350 [K]</td>
</tr>
<tr>
<td>( V ) - CSTR volume</td>
<td>100 [l]</td>
</tr>
<tr>
<td>( hA ) - Heat transfer term</td>
<td>( 7 \times 10^5 ) [cal/min/K]</td>
</tr>
<tr>
<td>( k_0 ) - Reaction rate constant</td>
<td>( 7.2 \times 10^{10} ) [min(^{-1})]</td>
</tr>
<tr>
<td>( E/R ) - Activation energy term</td>
<td>( 1 \times 10^4 ) [K]</td>
</tr>
<tr>
<td>( -\Delta H ) - Heat of reaction</td>
<td>( -2 \times 10^5 ) [cal/mol]</td>
</tr>
<tr>
<td>( \rho, \rho_c ) - Liquid densities</td>
<td>( 1 \times 10^3 ) [g/l]</td>
</tr>
<tr>
<td>( C_p, C_{pc} ) - Specific heats</td>
<td>( 1 ) [cal/g/K]</td>
</tr>
</tbody>
</table>
given to the closest center. Depending on the application, the \( \eta \) parameter changes in \( \eta \in (1, \infty) \). In many cases, these changes can reduce efficiency of the T-S regressor which is initialized by FCM. Currently many robust algorithms are being suggested to address this concern, e.g. (Ji et al., 2014). For the CSTR, \( \eta = 2 \) was seen as an appropriate value. However the results in Table 5 show that the NUFCA can reduce sensitivity to the \( \eta \), even when \( \eta \) is changed. The sensitivity parameter \( \alpha \) is defined as follows:

\[
\alpha(\hat{\eta}) = \frac{|J(\eta) - J(\hat{\eta})|}{J(\eta)} \times 100.
\]

where \( J \) is fitness evaluation, \( \hat{\eta} = 2, \ldots, 10 \), and the \( \eta = 2 \) which was found as the appropriate value for the WWTP application.

Results that show the improvement of performance of the NFCRMA when used in integration with the NUFCA are presented in Figure 5, where MSE\(_k\) is the single-sample MSE, i.e. it is the MSE considering only sample \( k \).

### 6.2.2. Adaptive Predictive Fuzzy Control of a Simulated CSTR

The model learned by Algorithm 1 (Sec. 3) is used to initialize the prediction model of the adaptive fuzzy GPC controller (Sec. 5). The following controller parameters were chosen: \( N_p = 150 \), \( N_u = 1 \), \( \lambda = 0.05 \), \( \sigma = 5 \), \( \rho = 0.999 \), \( \varphi_i = 1 \), \( \tau_i = \nu_i = 1 \times 10^{-9} \), for \( i = 1, \ldots, c \). A zero-mean Gaussian white noise with a variance of \( 2e^{-6} \) ([mol/1/min]\(^2\)) was also applied as a stochastic disturbance \( \vartheta(t) \). For a better study, two cases were tested.

<table>
<thead>
<tr>
<th>Method</th>
<th>( 1/\text{MSE} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPLS (Dayal &amp; MacGregor, 1997)</td>
<td>( 7.0871 \times 10^4 )</td>
</tr>
<tr>
<td>NFCRMA (Li et al., 2009)</td>
<td>( 7.3445 \times 10^6 )</td>
</tr>
<tr>
<td>ILLSA (Kadlec &amp; Gabrys, 2011)</td>
<td>( 3.9814 \times 10^5 )</td>
</tr>
<tr>
<td>HGA (Mendes et al., 2012)</td>
<td>( 5.2882 \times 10^5 )</td>
</tr>
<tr>
<td>Proposed Method (Algorithm 1)</td>
<td>( 9.2621 \times 10^7 )</td>
</tr>
</tbody>
</table>
Table 5: Comparison results on the test dataset for the CSTR plant (tests were run on a PC with a Intel(R) core (TM) i7-2600 and CPU 3.4GHz).

<table>
<thead>
<tr>
<th>Methods</th>
<th>Initialization type</th>
<th>1/MSE</th>
<th>Training time [min]</th>
<th>Sensitivity(α)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RLS (Kulhavý, 1987)</td>
<td>Randomly</td>
<td>$5.6742 \times 10^5$</td>
<td>6.852</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>FCM</td>
<td>$8.7738 \times 10^6$</td>
<td>5.931</td>
<td>0-64 [%]</td>
</tr>
<tr>
<td></td>
<td>NUFCMA (Li et al., 2009)</td>
<td>$9.2621 \times 10^5$</td>
<td>5.673</td>
<td>0-14 [%]</td>
</tr>
<tr>
<td></td>
<td>Randomly</td>
<td>$7.3406 \times 10^6$</td>
<td>12.646</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>FCM</td>
<td>$8.6602 \times 10^6$</td>
<td>9.6722</td>
<td>0-46 [%]</td>
</tr>
<tr>
<td></td>
<td>NUFCMA</td>
<td>$3.3565 \times 10^5$</td>
<td>8.838</td>
<td>0-11 [%]</td>
</tr>
<tr>
<td>HGA (Mendes et al., 2012)</td>
<td>Randomly</td>
<td>$1.0088 \times 10^2$</td>
<td>22.152</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>FCM</td>
<td>$4.6195 \times 10^5$</td>
<td>16.446</td>
<td>0-38 [%]</td>
</tr>
<tr>
<td></td>
<td>NUFCMA</td>
<td>$5.2882 \times 10^2$</td>
<td>13.358</td>
<td>0-9 [%]</td>
</tr>
</tbody>
</table>

Case 1: The reference input $r(t) [\text{mol/l}]$ is

$$r(t) = \begin{cases} 0.1, & 0 < t \leq 5 [\text{min}], \\ 0.08, & 5 [\text{min}] < t \leq 10 [\text{min}], \\ 0.09, & 10 [\text{min}] < t \leq 15 [\text{min}], \\ 0.08, & 15 [\text{min}] < t \leq 20 [\text{min}], \\ 0.1, & 20 [\text{min}] < t \leq 25 [\text{min}], \\ \end{cases}$$

and the load disturbance is defined as a change of the process flow rate $q$, where $q = 110$ for $13 [\text{min}] \leq t \leq 17 [\text{min}]$.

Case 2: In this case, the reference input is constant $r(t) = 0.1 [\text{mol/l}]$ for $0 [\text{min}] < t \leq 25 [\text{min}]$ and a disturbance is defined as a change of the feed concentration to $C_{A0} = 0.97 [\text{mol/l}]$ and also a change of the inlet coolant temperature to $T_{o0} = 345 [\text{K}]$, where both changes are in effect during $13 [\text{min}] \leq t \leq 17 [\text{min}]$.

In both cases, a comparison between the proposed AFGPC and the classical GPC (Camacho & Bordons, 1998) are presented. As the best result for the classical GPC, $N_p = 200$, $N_q = 1$, and $\lambda = 0.5$ were chosen. The linear model parameters used in the GPC controller were obtained with the Reaction Curve Method from (Camacho & Bordons, 1998). From the results presented in Figure 6 for Case 1, and Figure 7 for Case 2, it can be seen that the proposed controller is able to adequately (attain and) control the system output at the desired reference $r(t)$. When the load disturbance is applied at $13 [\text{min}] \leq t \leq 17 [\text{min}]$, there is an undershoot at $t = 13 [\text{min}]$ and an overshoot at $t = 17 [\text{min}]$ in the system response. The results revealed that the controller shows a robust performance with this disturbance. Also, comparing with the classical GPC, the proposed AFGPC has attained a faster response when the reference signal was changed. By the results, it is concluded that the proposed controller methodology can control the process using only a dataset of the process to initialize the T-S fuzzy model.

Figure 6: (a) The results of the proposed controller for Case 1 with Algorithm 1 and AFGPC in the presence of disturbances in the CSTR process and the comparison with the classical GPC; and (b) the respective applied command signals.
Time, kT/60 [min]

\[ y(k) - GPC \]
\[ y(k) - AFGPC \]

Figure 7: (a) The results of the proposed for Case 2 controller with Algorithm 1 and AFGPC in the presence of disturbances in the CSTR process and the comparison with the classical GPC; and (b) the respective applied command signals.

6.3. Real-World Control of Two Coupled DC Motors

The experimental system consists of two similar DC motors coupled by a shaft (Figure 8), where the first motor acts as an actuator, while the second motor is used as a generator and to produce nonlinearities and/or a time-varying load. The system exhibits noise, parasitic electromagnetic effects, friction and other phenomena commonly encountered in practical applications, that make the control task more difficult. The voltage command signal to the DC motor is in the range of \([0, 12] [\text{V}]\). The proposed control methodology runs on a PC that communicates by OPC\(^1\) to a PLC\(^2\) (ControlLogix L55 expanded with an analog I/O module for signal conditioning). The PLC provides the voltage command signal to the DC motor through the signal conditioning circuit. The velocity units are \([\text{pp} / (0.25 \text{ seg})]\) (pulses per 250 milliseconds). The generator has an electrical load composed of 2 lamps connected in parallel. When the lamps are connected in the generator circuit, the electrical load to the generator is increased (load resistance is decreased), and consequently the mechanical load that the generator applies to the motor also increases. Thus, it is possible to change the mechanical load to the motor, and consequently change its model. The main goal is to perform a velocity control where the load of the DC motor can be changed.

6.3.1. Identification

To identify the experimental setup, a dataset was constructed. The dataset was obtained by applying to the motor the control signal represented in Figure 9a. The variables chosen for the dataset were the first four delayed versions of the velocity \(y(k - 1), y(k - 2), y(k - 3), y(k - 4)\), and the command signal and its first three delayed versions \(u(k), u(k - 1), u(k - 2), u(k - 3)\), where \(k\) is the sample time. Applying the proposed methodology, the optimal number of clusters was calculated as \(c = 12\). The degree of fuzziness was chosen as \(\eta = 2\).

Numerical results comparing the performance of the proposed method and the works RPLS (Dayal & MacGregor, 1997), ILLSA (Kadlec & Gabrys, 2011), HGA (Mendes et al., 2012), and RLS (Kulhavý, 1987) are presented in Table 6. Figure 9b shows the comparison of the velocity values of the motor obtained by the proposed methodology (Algorithm 1), by the RLS, and the real/observed velocity values. It can be seen that the modeling of the velocity by the proposed methodology is accurate and better than the modeling obtained by the RLS method which uses random initialization.

<table>
<thead>
<tr>
<th>Method</th>
<th>1/MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>RLS (Kulhavý, 1987)</td>
<td>0.0523</td>
</tr>
<tr>
<td>RPLS (Dayal &amp; MacGregor, 1997)</td>
<td>0.02410</td>
</tr>
<tr>
<td>ILLSA (Kadlec &amp; Gabrys, 2011)</td>
<td>0.0197</td>
</tr>
<tr>
<td>HGA (Mendes et al., 2012)</td>
<td>0.0158</td>
</tr>
<tr>
<td>Proposed method (Algorithm 1)</td>
<td>0.1153</td>
</tr>
</tbody>
</table>

Table 6: Comparison results on the test dataset for the DC motor.

---

\(^1\)OLE (Object Linking and Embedding) for Process Control.
\(^2\)Programmable Logic Controller.
6.3.2. Adaptive Predictive Fuzzy Control

The model learned by Algorithm 1 is used to initialize the prediction model of the adaptive fuzzy GPC controller. The following controller parameters were chosen by the user: $N_p = 8$, $N_u = 1$, $\lambda = 25$, $d = 0$, $\rho = 0.93$, $\varphi_i = 1$, $\tau_i = 1 \times 10^{-3}$, $\psi_i = 1 \times 10^{-6}$, for $i = 1, \ldots, c$. The reference input $r(t) \text{[pp/(0.25 seg)]}$ is

$$r(k) = \begin{cases} 
100, & 0 < k \leq 120, \\
150, & 120 < k \leq 200, \\
130, & 200 < k \leq 360, \\
120, & 360 < k \leq 560, \\
100, & 560 < k \leq 640,
\end{cases}$$

and the load disturbance (lamps switched on) is applied at $260 \leq k \leq 410$. Performance of both the proposed AFGPC controller and the classical GPC controller (Camacho & Bordons, 1998) is presented in Figures 10a and 10b. For the classical GPC $N_p = 8$, $N_u = 1$, and $\lambda = 25$ were chosen. The linear model parameters used in the GPC controller were obtained with the Reaction Curve Method from Camacho & Bordons (1998). From the results presented in Figures 10a and 10b, it can be seen that the proposed controller is able to adequately (attain and) control the system output at the desired reference $r(k)$. When the load disturbance is applied at $260 \leq k \leq 410$, there is an undershoot at $k = 260$ and an overshoot at $k = 410$ in the system response. As can be seen the controller shows a robust performance against this disturbance. The results can imply that using the proposed and combined Algorithm 1-AFGPC controller methodology can lead to the adequate control of the process just based on a dataset of the process to initialize the T-S fuzzy model. Furthermore, the result show that comparing with the classical GPC, the proposed AFGPC causes the output $y(k)$ to follow the reference sig-
nal r(k) with less oscillation, and the AFGPC command signal u(k) also has less oscillation when comparing with the GPC command. Note that when exposing the plant to constraints on the command variations \( \Delta u(k) \), large scales of unexpected variations in the command signal or output can even significantly deteriorate the closed-loop performance of the plant, but the AFGPC is able to mitigate such problems. The design of the proposed AFGPC is based on the adaptive model, where the consequent parameters of the T-S fuzzy model are updated whenever a new input data sample becomes available. With this strategy a better estimation performance, which is more close to the performance of the real plant, can be obtained. Consequently as it can be seen in the Figures 10a and 10b, an input command signal and output, both with less variations can be expected which is desirable characteristics to improve the performance of the control system, as well as it can permit to reduce costs.

7. Summary and Outlook

An algorithm was proposed to identify a T-S fuzzy model to approximate unknown nonlinear processes based on input/output data. The method has been proposed to identify the structure and parameters of the model: the fuzzy rules, and the antecedent parameters of fuzzy membership functions are automatically learned from system data. The proposed hybrid identification methodology can be considered for application in problems such as the design of data-driven soft sensors, or in model based predictive controller. A recursive least squares method with adaptive directional forgetting is used for online adaptation of the T-S fuzzy model. In this paper, the learned model was integrated with a fuzzy GPC controller. The integration of the proposed adaptive identification method with the GPC results in an effective adaptive predictive fuzzy control methodology. To validate and demonstrate the performance and effectiveness of the proposed algorithms, they were tested on the identification problem of estimation of the flour concentration in the effluent of a real-world wastewater treatment system; and on the control of a simulated continuous stirred tank reactor (CSTR), and on the control of a real-world experimental setup composed of two coupled DC motors. In general, the presented identification results, when comparing with other methods, have shown that the adjustment of model parameters by the proposed method can provide advantages of better performance in: (1) attaining a lower MSE, (2) a lower number of fuzzy rules, (3) using lower computation time for offline system identification comparing with some recent proposed identification methods which follow an evolutionary procedure. Moreover, in results with the predictive control process, the proposed AFGPC comparing with the classical GPC (Camacho & Bordons, 1998) has performed with a faster response and less oscillation. As a future work, stability analysis of the proposed AFGPC will be investigated. Results have shown that the proposed controller methodology can control the process using only a dataset of the process to initialize the adaptive T-S fuzzy model.

Funding

This work was supported by Project SCIAD/2011/21531 co-financed by QREN, in the framework of the “Mais Centro - Regional Operational Program of the Centro”, and by the European Union through the European Regional Development Fund (ERDF).

Saeid Rastegar and Jérôme Mendes were supported by Fundação para a Ciência e a Tecnologia (FCT) under grants SFHR/BD/89186/2012 and SFHR/BD/63383/2009, respectively.

References


