Evolving Systems (2011) 2:15–24
DOI 10.1007/s12530-010-9025-7

ORIGINAL PAPER

Recursive clustering based on a Gustafson–Kessel algorithm

Dejan Dovžan · Igor Škrjanc

Received: 4 October 2010 / Accepted: 2 December 2010 / Published online: 23 December 2010

Abstract In this paper an on-line fuzzy identification of Takagi Sugeno fuzzy model is presented. The presented method combines a recursive Gustafson–Kessel clustering algorithm and the fuzzy recursive least squares method. The on-line Gustafson–Kessel clustering method is derived. The recursive equations for fuzzy covariance matrix, its inverse and cluster centers are given. The use of the method is presented on two examples. First example demonstrates the use of the method for monitoring of the waste water treatment process and in the second example the method is used to develop an adaptive fuzzy predictive functional controller for a pH process. The results for the Mackey–Glass time series prediction are also given.

Keywords Recursive fuzzy clustering · Recursive fuzzy identification · Clustering · Online recursive identification · Recursive Gustafson–Kessel clustering

1 Introduction

Takagi–Sugeno models (Takagi and Sugeno 1985) are a powerful practical engineering tool for modeling and control of complex systems. Due to the fuzzy regions (clusters), the nonlinear system is decomposed into a multi-model structure consisting of linear models (Johanson and Murray-Smith 1997). This enables the T–S fuzzy model to approximate virtually any nonlinear system within a required accuracy, provided that enough regions are given (Hwang and Chang 2007).

There are a number of off-line methods for identification of fuzzy model (Kukolj and Levi 2004; Kim et al. 1997; Shing and Jang 1993; Kasabov 1996). In recent years there has been an increased interest in an on-line nonlinear model identification that employ fuzzy logic and its combination with the neural networks (Angelov and Filev 2004; Kasabov and Song 2002; Kim et al. 2005; Leng et al. 2004; Kasabov 2001, 1998a, b; Qiao and Wang 2008; Xu et al. 1993; Hai-Jun et al. 2006; Juang and Lin 1998; Wang et al. 2008; Lin et al. 2001; Wu and Er 2000; Wu et al. 2001; Lin 1995; Tzafestas and Zikidis 2001; Azeem et al. 2003). The methods employ different clustering algorithms for example Widrow–Hoffs LMS algorithm, evolving clustering method (Kasabov and Song 2002), N-first nearest neighborhood heuristic, error reducing and structure-evolving mechanism, modified mountain clustering (Azeem et al. 1999), Kohen's feature maps and other algorithms. The algorithms use mostly Gaussian or triangular membership functions.

Probably one of the most promising methods for on-line identification of the Takagi–Sugeno fuzzy model is the evolving Takagi–Sugeno model (eTS) (Angelov and Filev 2004). The method is a further development of the evolving rule-based models (eR) (Angelov 2002). It uses the informative potential of a new data sample to update the rule base. To estimate the clusters the recursive clustering algorithm (Angelov 2004) based on subtractive clustering (Chiu 1994) is used. This is an improved version of mountain clustering (Yager and Filev 1993). The width of the membership functions in the original eTS method was fixed, but the method was improved so that the width is also automatically set [exTS (Angelov and Zhou 2006) and eTS+ (Angelov 2010)]. The improved version can

D. Dovžan (✉) · I. Škrjanc
Faculty of Electrical Engineering, Tržaška 25,
Ljubljana, Slovenia
e-mail: adejan.dovzan@fe.uni-lj.si

I. Škrjanc
e-mail: bigor.skrjanc@fe.uni-lj.si

therefore detect different shapes of clusters without inverting the fuzzy covariance matrix. The exTS uses local scatter measurement over the input space that resembles the variance.

Recently an extension of the Gustafson–Kessel clustering algorithm for evolving data stream clustering was published as a chapter in a book (Filev and Georgieva 2010). The algorithm is also based on a GK off-line clustering algorithm as ours, but the adaptation of centers and the calculation of the fuzzy covariance matrix are different. A recursive version of Gath-Geva clustering (Soleimani-B et al. 2010) algorithm was also published recently.

The idea behind the proposed algorithm is similar as in (Angelov and Filev 2004). The difference is that they based the algorithm on an off-line subtractive clustering algorithm and the proposed one is based on a Gustafson–Kessel off-line algorithm. Both algorithm use the Gaussian membership function, but eTS is able to add new cluster centers as apposed to the proposed, where the number of clusters are fixed.

The proposed algorithm is developed for on-line identification and adaptation of the fuzzy model. It is developed to be used in a fuzzy predictive functional controller (Škrjanc and Matko 2000) to control nonlinear dynamic systems, where the nonlinearity is time dependent. It can also be used for a fault detection of a non-linear process, where the dynamic of the system changes, such as waste water treatment process. For the basis we chose the *GK* clustering, because is well known, widely used for constructing fuzzy models that are used for model predictive control and gives good results. The recursive equations are derived directly from equations of the off-line algorithm.

The paper is organized in the following fashion. First the recursive identification method is developed based on GK and least squares algorithm. The use of the method is then demonstrated on an example of monitoring of waste water treatment process and on the control of the pH process. Then the results for Mackey–Glass (M–G) time series prediction are given. At the end some conclusions are made.

2 Fuzzy *c*-means and Gustafson–Kessel clustering

Fuzzy *c*-means (FCM) is probably one of the most known off-line clustering algorithms (Bezdek 1981). It is based on a minimization of the fuzzy *c*-means objective function. From the minimization problem the equations for the membership degrees (μ_i) and cluster centers (v_i) are obtained:

$$\mu_i(k) = \left(d_{ik}^2 \sum_{j=1}^{c} \left(\frac{1}{d_{jk}^2} \right)^{\frac{1}{\eta-1}} \right)^{-1} \tag{1}$$

$$v_i = \frac{\sum_{k=1}^{n} \mu_i^{\eta}(k) x(k)}{\sum_{k=1}^{n} \mu_i^{\eta}(k)}, \tag{2}$$

where d_{ik} is the Euclidian distance between the observation $x(k)$ (data vector $x(k) = [x_1(k), \ldots, x_m(k)]$) and the cluster centroid v_i. The FCM algorithm is based on assumption that the clusters are spherical shaped. In number of real problems and dynamical systems the clusters are of different shapes and with different orientations in the data space. To detect different geometrical shapes in data sets Gustafson and Kessel (1979) extended the FCM algorithm by employing an adaptive distance norm for each cluster. Each *i*th cluster has its own norm-inducing matrix A_i which affects the distance norm. Euclidan norm in the FCM is replaced with Mahalanobis distance norm given as follows:

$$D_{ik,A_i}^2 = (x(k) - v_i)^T A_i (x(k) - v_i), \tag{3}$$

where A_i is obtained from fuzzy covariance matrix F_i defined as:

$$F_i = \frac{\sum_{k=1}^{n} \mu_i^{\eta}(k)(x(k) - v_i)(x(k) - v_i)^T}{\sum_{k=1}^{n} \mu_i^{\eta}(k)}, \tag{4}$$

$$A_i = [\rho_i \; det(F_i)]^{1/p} \; F_i^{-1}, \tag{5}$$

where p is the number of measured variables and ρ_i is a cluster volume, which is usually set to 1 for each cluster. This allows the algorithm to find clusters of approximately equal volumes.

3 Recursive GK clustering

When the behavior of the process that generates the observed data changes during the time, the clustering should be done recursively to obtain the clusters that describe the current behavior. The recursive method can also be used for constructing the fuzzy model on-line.

3.1 The recursive center calculation

To develop the recursive fuzzy clustering algorithm we will first define the cluster centroid vector $v_i^T = [v_{i1}, \ldots, v_{i_m}]$ according to the current observation, i.e., the weighted mean of the data according to the current membership degrees. This introduces the notation $v_i(r)$, which means the cluster centroid at the time instant r that is obtained by weighting with the current membership

degrees. The cluster centroid in the next observation is denoted as

$$v_i(r+1) = \frac{\sum_{k=1}^{r+1} \mu_i^{\eta}(k)x(k)}{\sum_{k=1}^{r+1} \mu_i^{\eta}(k)} = \frac{\sum_{k=1}^{r} \mu_i^{\eta}(k)x(k) + \mu_i^{\eta}(r+1)x(r+1)}{\sum_{k=1}^{r} \mu_i^{\eta}(k) + \mu_i^{\eta}(r+1)} \quad (6)$$

where $\mu_i(k)$, $k = 1, \ldots, r+1$ denotes the membership degree of the observation vector $x(k)^T = [x_1(k), \ldots, x_m(k)]$, $k = 1,..., r+1$ to the cluster i at the time instant k. Introducing the relation between the old cluster centroid and a new one as follows:

$$v_i(r+1) = v_i(r) + \Delta v_i(r+1) \quad (7)$$

and taking into account Eq. 6 the Eq. 11 is obtained:

$$v_i(r+1) = \frac{\sum_{k=1}^{r} \mu_i^{\eta}(k)x(k) / \sum_{k=1}^{r} \mu_i^{\eta}(k)}{\sum_{k=1}^{r+1} \mu_i^{\eta}(k) / \sum_{k=1}^{r} \mu_i^{\eta}(k)} + \frac{\mu_i^{\eta}(r+1)x(r+1)}{\sum_{k=1}^{r} \mu_i^{\eta}(k) + \mu_i^{\eta}(r+1)} \quad (8)$$

$$v_i(r+1) = \frac{v_i(k)\sum_{k=1}^{r} \mu_i^{\eta}(k)}{\sum_{k=1}^{r+1} \mu_i^{\eta}(k)} + \frac{\mu_i^{\eta}(r+1)x(r+1)}{\sum_{k=1}^{r} \mu_i^{\eta}(k) + \mu_i^{\eta}(r+1)} \quad (9)$$

$$v_i(r+1) = v_i(k) - \frac{v_i(k)\mu_i^{\eta}(r+1)}{\sum_{k=1}^{r+1} \mu_i^{\eta}(k)} + \frac{\mu_i^{\eta}(r+1)x(r+1)}{\sum_{k=1}^{r} \mu_i^{\eta}(k) + \mu_i^{\eta}(r+1)} \quad (10)$$

$$\Delta v_i(r+1) = \frac{\mu_i^{\eta}(r+1)(x(r+1) - v_i(r))}{\sum_{k=1}^{r} \mu_i^{\eta}(k) + \mu_i^{\eta}(r+1)} \quad (11)$$

The term in the denominator of equation can be denoted as $s_i(k+1) \in \mathbb{R}^c$ and calculated as:

$$s_i(r+1) = s_i(r) + \mu_i^{\eta}(r+1) \quad (12)$$

where $s_i(r)$ is defined as follows

$$s_i(r) = \sum_{k=1}^{r} \mu_i^{\eta}(k) \quad (13)$$

Introducing the forgetting factor, Eq. 12 can be rewritten as:

$$s_i(r+1) = \gamma_v\, s_i(r) + \mu_i^{\eta}(r+1). \quad (14)$$

The parameter γ_v, $(0 \le \gamma_v \le 1)$ denotes the forgetting factor of a past observation, i.e., the forgetting factor of the past membership degrees. The $\Delta v_i(r+1)$ can now be written as:

$$\Delta v_i(r+1) = \frac{\mu_i^{\eta}(r+1)(x(r+1) - v_i(r))}{s_i(r+1)} \quad (15)$$

The current membership degree $\mu_i(r+1)$ is next defined as follows:

$$\mu_i(r+1) = \left(D^2_{i,r+1,A_i} \sum_{j=1}^{c} \left(\frac{1}{D^2_{j,r+1,A_j}} \right)^{\frac{1}{\eta-1}} \right)^{-1} \quad (16)$$

where $D^2_{i,r+1,A_i}$ defines the quadratic distance from the cluster centroid as follows:

$$D^2_{i,r+1,A_i} = (x(r+1) - v_i(r))^T A_i (x(r+1) - v_i(r)),\ 1 \le i \le c. \quad (17)$$

The matrix A_i is calculated from fuzzy covariance matrix (Eq. 4) at each step. The equation for the fuzzy covariance matrix must be rewritten in a form suitable for a recursive calculation. The equation for the fuzzy covariance matrix can be written in a following form:

$$F_i(r) = \frac{\sum_{k=1}^{r} \mu_i^{\eta}(k)\left(x(k) - v_i^r\right)\left(x(k) - v_i^r\right)^T}{\sum_{k=1}^{r} \mu_i^{\eta}(k)} \quad (18)$$

where v_i^r stands for the centroid vector of the ith cluster calculated for the set of r samples. The fuzzy covariance matrix at the next sample $(r+1)$ can be expressed as follows

$$F_i(r+1) = \frac{\sum_{k=1}^{r} \mu_i^{\eta}(k)\left(x(k) - v_i^{r+1}\right)\left(x(k) - v_i^{r+1}\right)^T}{\sum_{k=1}^{r} \mu_i^{\eta}(k) + \mu_i^{\eta}(r+1)} + \frac{\mu_i^{\eta}(r+1)\left(x(r+1) - v_i^{r+1}\right)\left(x(r+1) - v_i^{r+1}\right)^T}{\sum_{k=1}^{r} \mu_i^{\eta}(k) + \mu_i^{\eta}(r+1)} \quad (19)$$

where v_i^{r+1} stands for the centroid vector of the ith cluster calculated for the set of $r+1$ samples.

Taking into account Eq. 18, introducing it into Eq. 19, and using Eq. 12 and Eq. 13, the following recursive expression for the fuzzy clustering matrix approximation is obtained

$$F_i(r+1) = \gamma_c \frac{s_i(r)}{s_i(r+1)} F_i(r) + \frac{\mu_i^{\eta}(r+1)}{s_i(r+1)} \cdot (x(r+1) - v_i(r+1))(x(r+1) - v_i(r+1))^T, \quad (20)$$

where γ_c was introduced as forgetting factor of fuzzy covariance matrix. For recursive Gustafson Kessel method we also need an inverse of the fuzzy covariance matrix. The calculation of the inverse is a slow operation. Therefore we used the Woodbury inverse lemma to gain the recursive calculation of the inverse fuzzy matrix:

$$F_i^{-1}(r+1) = \frac{A - B}{C} \quad (21)$$

$$A = \frac{1}{\gamma_c} \frac{s_i(r+1)}{s_i(r)} F_i(r)^{-1} \quad (22)$$

$$B = \frac{1}{\gamma_c}\frac{s_i(r+1)}{s_i(r)}F_i^{-1}\frac{\mu_i^{\eta}(r+1)}{s_i(r+1)}(x(r+1)-v_i(r+1)) \cdot (x(r+1)-v_i(r+1))^T\frac{1}{\gamma_c}\frac{s_i(r+1)}{s_i(r)}F_i^{-1} \quad (23)$$

$$C = 1 + (x(r+1)-v_i(r+1))^T\frac{1}{\gamma_c}\frac{s_i(r+1)}{s_i(r)} \cdot F_i^{-1}\frac{\mu_i^{\eta}(r+1)}{s_i(r+1)}(x(r+1)-v_i(r+1))) \quad (24)$$

Now the inversion of the fuzzy covariance matrix is done only at the initialization step.

3.2 Applying the recursive least squares

The centers of the fuzzy clusters and their distribution are used to define the new membership functions and using the recursive least squares method the fuzzy model is obtained. Using the projection of the cluster onto the independent variables, the input membership functions are obtained. Here we are assuming that the first $m-1$ measured variables represent the input variables and the last mth variable in the observation vector ($x(k)$) represents the output.

In our case we used the Gaussian membership functions. The membership function of the ith cluster and the jth component of $x(k)$ is defined as

$$\mu_i(x_j(k)) = e^{-\frac{(x_j(k)-v_{ij}(k))(x_j(k)-v_{ij}(k))^T}{2\mu_m\sigma_{ij}^2(k)}}, \quad i=1,\ldots,c,\ j=1,\ldots,m-1 \quad (25)$$

where v_{ij} is the jth component of ith cluster center, x_j is the jth component of the observation vector and σ_{ij} is the variance (the jth diagonal element of matrix F_i). The overlapping factor μ_m defines the overlapping of the membership functions. Usually it can be set to one so that the membership functions resemble the Gaussian distribution. The $m-1$ input variables define the input hyperspace. The subspaces in this hyperspace are defined as the Cartesian product of the subspaces. These imply the definition of the membership degree in each subspace as the product of the membership degrees as follows:

$$\beta_i(k) = \prod_{j=1}^{m-1}\mu_i(x_j(k)) \quad (26)$$

The fuzzy recursive least squares algorithm (Goodwin and Sin 1984, Kasabov and Song 2002, Angelov and Filev 2004) is then used to estimate the local-linear sub-models parameters. We can use version for global optimization case:

$$\psi_i^T(k+1) = \beta_i(k)[1,\ x_1(k),\ x_2(k),\ \ldots, x_{m-1}(k)] \quad (27)$$

$$y_i(k) = \beta_i(k)\ x_m(k) \quad (28)$$

$$P_i(k+1) = \frac{1}{\lambda_r}\left(P(k) - \frac{P_i(k)\psi_i(k+1)\psi_i^T(k+1)P_i(k)}{\lambda_r + \psi_i^T(k+1)P_i(k)\psi_i(k+1)}\right) \quad (29)$$

$$\theta_i(k+1) = \theta_i(k) + P_i(k)\psi_i(k+1)\big(y_i(k) - \psi_i^T(k+1)\theta_i(k)\big) \quad (30)$$

or for local optimization case:

$$\psi_i^T(k+1) = [1,\ x_1(k),\ x_2(k),\ \ldots,\ x_{m-1}(k)] \quad (31)$$

$$y_i(k) = x_m(k) \quad (32)$$

$$P_i(k+1) = \frac{1}{\lambda_r}\left(P(k) - \frac{\beta_i(k)P_i(k)\psi_i(k+1)\psi_i^T(k+1)P_i(k)}{\lambda_r + \beta_i(k)\psi_i^T(k+1)P_i(k)\psi_i(k+1)}\right) \quad (33)$$

$$\theta_i(k+1) = \theta_i(k) + P_i(k)\psi_i(k+1)\beta_i(k)\big(y_i(k) - \psi_i^T(k+1)\theta_i(k)\big) \quad (34)$$

where λ_r stands for the exponential forgetting factor, which should be set between 0.98 and 1 to deal with time-varying processes (Åström and Wittenmark 1995), P_i stands for the covariance matrix, which is set to $P_i(0) = 10^2 \sim 10^5 I$, $I \in \mathbb{R}^{m\times m}$, and θ_i represents the parameters of the ith local model.

3.3 The steps of the algorithm

The algorithm can be described in ten steps that are made at each time instant k and initialization step. The clustering algorithm is from step one to step nine. Step nine represents the fuzzy recursive least squares.

1. Step 0: Initialization:
   - define the number of clusters (c), the fuzziness and overlapping (η, η_m), the forgetting factors ($\lambda_r, \gamma_v, \gamma_c$), number of measured variables (p) and the cluster volumes (ρ_i),
   - determine the initial value of P_i for $i = 1,\ldots,c$,
   - determine the initial F_i for $i = 1,\ldots,c$,
   - determine the initial centers: $v_i = x(k)$, for $i = k = 1,\ldots,c$ and membership degrees $\mu_i(x(k)) = 1$ for $i = k$, $\mu_i(x(k)) = 0$ for $i \neq k$, $i,k = 1,\ldots,c$,
   - calculate the initial s_i for $i = 1,\ldots,c$ from Eq. 13,
   - calculate the inverse of the fuzzy covariance matrix.
2. Step 1: calculate the matrix A_i from Eq. 5,
3. Step 2: calculate the distance D_i from Eq. 17,
4. Step 3: calculate the current membership degree from Eq. 16,
5. Step 4: calculate the $s_i(r+1)$ from Eq. 14,
6. Step 5: calculate the change of cluster centers Δv_i from Eq. 11,

7. Step 6: calculate the new centers from Eq. 7,
8. Step 7: calculate the new fuzzy covariance matrix $F_i(r+1)$ from equation Eq. 20 and inverse fuzzy covariance matrix from Eqs. 22 to Eq. 24,
9. Step 8: calculate the membership functions μ_i from Eq. 25,
10. Step 9: calculate the membership degrees β_i from Eq. 26,
11. Step 10: apply the recursive least squares Eqs. 27–30 or Eqs. 31–34 and return to Step 1.

Step 0 is executed only once in the procedure. The first three tasks under step zero are done by user everything else is initialized from the algorithm itself. The steps from Step 1 to Step 10 are executed for the time instances $k \geq c+1$ for every new sample.

The parameters that have to be set in advance are: forgetting factors, number of clusters, fuzziness, overlapping factor, cluster volumes and number of measured variables. For time varying processes, the factors must be lower than one to ensure the forgetting. By lowering the forgetting factors the fluctuations of estimates will increase but the adaptation will be faster. The forgetting factors can be chosen with the help of the rule of thumb (Åström and Wittenmark 1995):

$$\lambda = 1 - 2/N, \tag{35}$$

where λ is the forgetting factor and N are the data samples, that affect the estimates.

The fuzziness factor and overlapping factor define the smoothness of the nonlinearity approximation. The higher they are the more smooth nonlinearity we get, but to high values cause bigger identification error and prediction error. The fuzziness factor is usually set $\eta = 2$ and the overlapping factor $\eta_m = 1$. The cluster volumes are usually se to one.

For the fuzzy covariance and least squares covariance matrix the initial values must be given. The matrix are usually initialized with the identity matrix as: $F_i(0) \approx I$ and $P_i(0) \approx 10^4 I$.

4 Fault detection based on a fuzzy model

In this example the proposed method is used to construct the fuzzy model based on a number of starting samples. The fuzzy model is then used to monitor the process and detect the fault. The process used in the example was a waste water treatment process.

Waste-water treatment plants are large nonlinear systems subject to large perturbations in flow and load, together with uncertainties concerning the composition of the incoming waste-water. The simulation benchmark has been developed to provide an unbiased system for comparing various strategies without reference to a particular facility. It consists of five sequentially connected reactors along with a 10-layer secondary settling tank. The plant layout, model equations and control strategy are described in detail on the web page (http://www.ensic.u-nancy.fr/costwwtp). In our example the phase where the waste-water is purified is monitored. After this phase the moving bed bio-film reactor is used. Schematic representation of simulation benchmark is shown in Fig. 1. The detection of sensor faults was applied to the simulation model where the following measurements were used to identify the fuzzy model: influent ammonia concentration in the inflow Q_{in} defined as $C_{NH4N_{in}}$, dissolved oxygen concentration in the first aerobic reactor tank $C^1_{O_2}$, dissolved oxygen concentration in the second aerobic reactor tank $C^2_{O_2}$ and the ammonia concentration in the second aerobic reactor tank $C_{NH4N_{out}}$. The fuzzy model was build to model the relation between the ammonia concentration in the second aerobic reactor tank and the other measured variables:

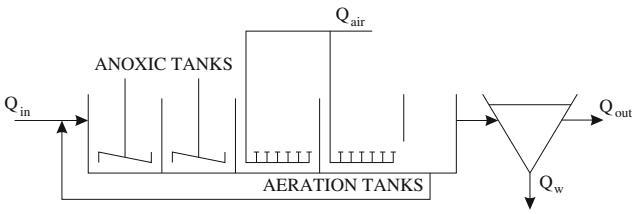

Fig. 1 Schematic representation of simulation benchmark

$$C_{NH4N_{\text{out}}}(k) = \mathcal{G}\left(C_{NH4N_{\text{in}}}(k), C^1_{\text{O}_2}(k), C^2_{\text{O}_2}(k)\right) \tag{36}$$

where $\mathcal{G}$ stands for nonlinear relation between measured variables. First 15,000 measurements (sampling time $T_s = 120$ s) were used to find the fuzzy clusters and to estimate the fuzzy model parameters. At the measurement 17,000 the slowly increasing sensor fault occur, which is than at sample 18,000 eliminated. This means that sensor to measure the ammonia concentration in the second aerobic reactor tank $C_{NH4N_{out}}$ is faulty. The signal with exponentially increasing value was added to the nominal signal. The whole set of measurements is shown in Fig. 2 and 3. The fuzzy model is obtained on the set of first 15,000 samples using the proposed method. The identification of the model is done on-line. At the sample 15,000 the model output is calculated for the starting period and the tolerance index are calculated. The fuzziness and overlapping were set to four and ten clusters were used. The forgetting factors for the first period were set to one (no forgetting). From the sample 15,000 the forgetting factors were set to 0.9998. The on-line adaptation is stopped when alarm is raised and started again after the end of alarm. The fuzzy

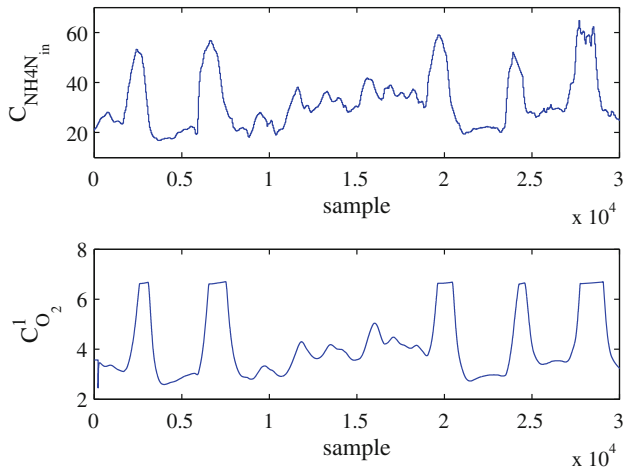

Fig. 2 The whole set of measurements. The influent ammonia concentration $C_{NH4N_{in}}$ and dissolved oxygen concentration in the first aerobic reactor tank $C^1_{O_2}$

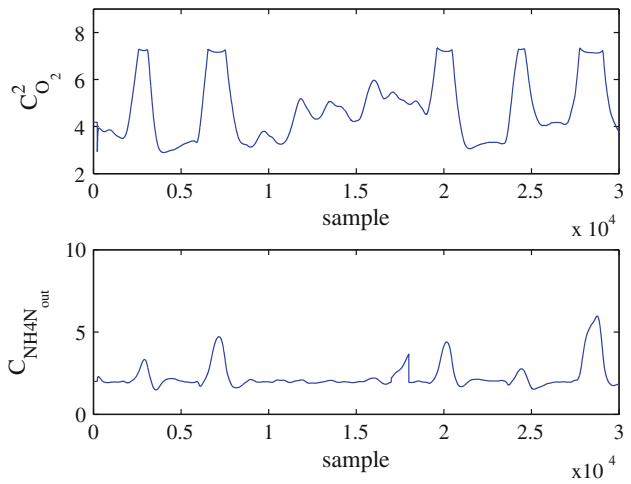

Fig. 3 The whole set of measurements. The dissolved oxygen concentration in the second aerobic reactor tank $C^2_{O_2}$ and the ammonia concentration in the second aerobic reactor tank $C_{NH4N_{out}}$

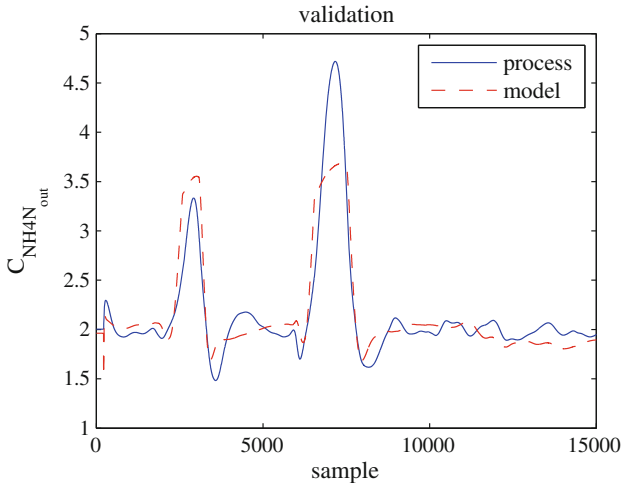

Fig. 4 The verification of fuzzy model where the fuzzy model output and the process output are shown

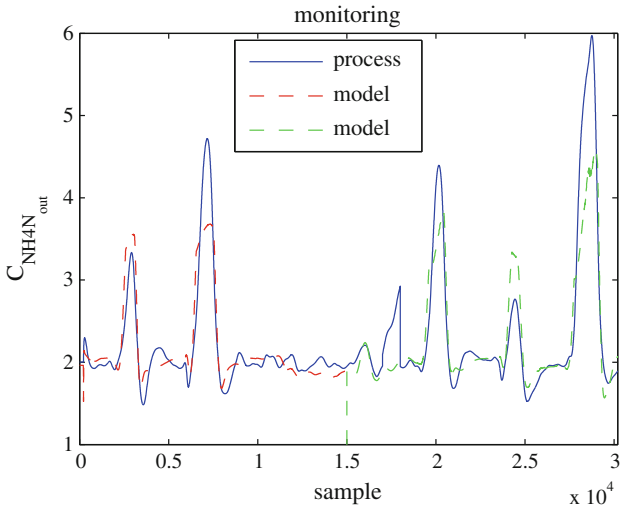

Fig. 5 The monitoring. The model output and the process output for the whole measurement set

model output $\hat{C}_{NH4N_{out}}$ and the process output $C_{NH4N_{out}}$ for the tuning period are shown in Fig. 4.

The fault detection index is defined as:

$$f = \left(\frac{C_{NH4N_{out}} - \hat{C}_{NH4N_{out}}}{\hat{C}_{NH4N_{out}}} \right)^2 \tag{37}$$

The fault tolerance index is defined as relative degree of maximal value of fault detection index in the identification or learning phase $f_{tol} = \gamma \max f$ where in our case $\gamma = 1.5$.

The algorithm calculated the value of fault detection index $f_{tol} = 0.156$. The fault which occur at the sample 17,000 is detected at the sample 17,425. The detection is delayed, but this is usual when the faults are slowly increasing. Figure 5 shows the process output and the model prediction for the whole experiment.

5 Fuzzy predictive functional control

This example demonstrates the use of the proposed recursive clustering method in a combination with the fuzzy predictive functional control algorithm. The method is used for adaptation of fuzzy model when the nonlinearity changes. This example the FPFC is used to control the pH process. The process was adopted from (Henson and Seborg 1994). The scheme of the process is

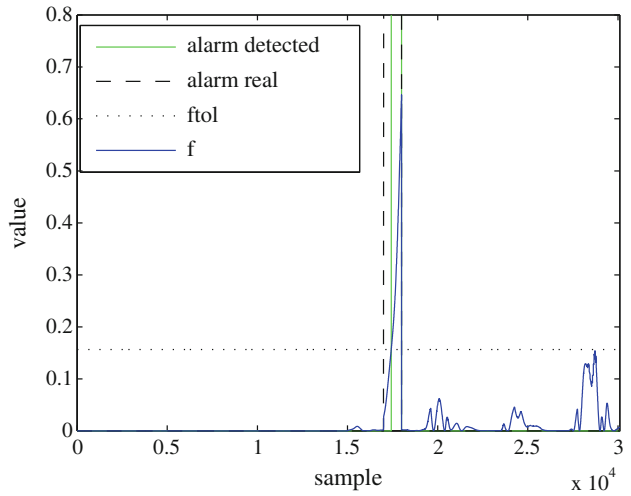

Fig. 6 The fault detection index, the fault tolerance index f and the actual and detected fault

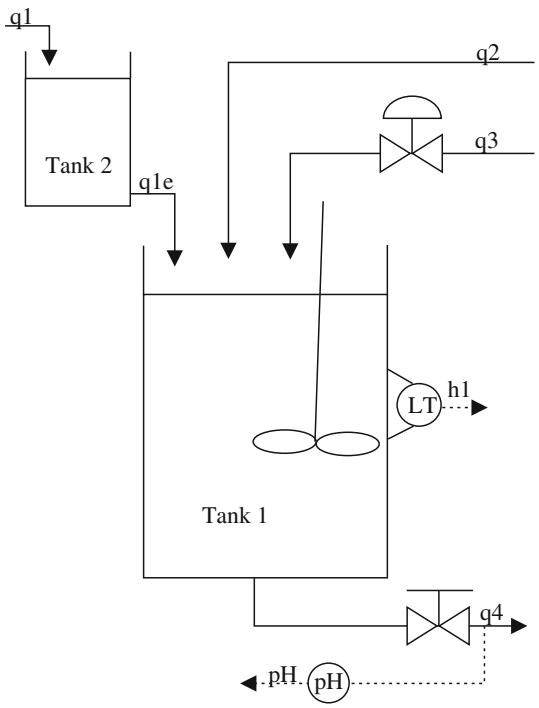

Fig. 7 The scheme of a pH process

shown on Fig. 7. The acid stream (q_1), buffer stream (q_2) and base stream (q_3) are mixed in tank 1. Prior to mixing the acid stream enters tank 2, which introduces an additional dynamics. The control valves control the acid and base flow. The tank level (h_1) and effluent pH (pH) are measured values. In this example the controlled variable is pH, manipulated by base flow rate (q_3), acid and buffer flows are considered to be unmeasured disturbances.

Equations that describe the process are following:

$$A_2\frac{dh_2}{dt} = q_1 - q_{1e} \tag{38}$$

$$q_{1e} = C_{v1}h_2^{0.5} \tag{39}$$

$$x = [W_{a4} \quad W_{b4} \quad h_1]^T \tag{40}$$

$$d = q_2 \tag{41}$$

$$u_a = q_3 \tag{42}$$

$$y = pH \tag{43}$$

$$f(x) = \left[\frac{q_1}{A_1x_3}(W_{a1} - x_1) \quad \frac{q_1}{A_1x_3}(W_{b1} - x_2) \times\frac{1}{A_1}(q_1 - C_{v4}(h_1 + z)^n)\right]^T \tag{44}$$

$$g(x) = \left[\frac{1}{A_1x_3}(W_{a3} - x_1) \quad \frac{1}{A_1x_3}(W_{b3} - x_2) \quad \frac{1}{A_1}\right]^T \tag{45}$$

$$p(x) = \left[\frac{1}{A_1x_3}(W_{a2} - x_1) \quad \frac{1}{A_1x_3}(W_{b2} - x_2) \quad \frac{1}{A_1}\right]^T \tag{46}$$

$$\dot{x} = f(x) + g(x)u_a + p(x)d \tag{47}$$

$$pK_{a2} = -log_{10}(K_{a2}) \quad pK_{a1} = -log_{10}(K_{a1}) \tag{48}$$

$$c(x,y) = x_1 + 10^{y-14} - 10^{-y} + x_2\frac{1 + 2\times 10^{y-pK_{a2}}}{1 + 10^{pK_{a1}-y} + 10^{y-pK_{a2}}} \tag{49}$$

$$c(x,y) = 0 \tag{50}$$

The parameters are given in Table 1. The change of the dynamics of the process is simulated by changing the buffer input stream q_2 to 0.8 ml/s and the acid stream q_1 to 10 ml/s.

To show the cluster center reposition the test in an open-loop was made with the random input signal. The regression vector was chosen as $x(i) = [u_a(i) \quad y(i)]^T$. Figure 8 shows the cluster centers and the input–output data distribution before the change. Figure 9 shows the cluster centers and the input–output data distribution after the change.

For the control the starting fuzzy model was identified on-line in an open loop using random generated input signal. This model was then used to control the process using fuzzy predictive functional controller. The control algorithm is detail described in (Dovžan and Škrjanc 2010) and (Škrjanc and Matko 2000). The forgetting factors were set to 0.999 and the resetting of covariance matrices was used (Åström and Wittenmark 1995). Figure 10 shows the control experiment. In the experiment the adaptive version of the FPFC and nonadaptive version are compared.

The control of the adaptive version is better than one with the nonadaptive. The disturbance rejection is faster Fig. 11 and has less overshoot.

Table 1 Operating conditions for pH process

$A_1 = 207\ \text{cm}^2$	$q_4 = 32.8$ ml/s
$A_2 = 42\ \text{cm}^2$	$W_{a1} = 3 \times 10^{-3}$ M
$z = 11.5\ \text{cm}^2$	$W_{b1} = 0$ M
$n = 0.607$	$W_{a2} = -0.03$ M
$K_{a1} = 4.47 \times 10^7$	$W_{b2} = 0.03$ M
$K_{a2} = 5.62 \times 10^{-11}$	$W_{a3} = -3.05 \times 10^{-3}$ M
$Kw = 1 \times 10^{14}$	$W_{b3} = 5 \times 10^{-5}$ M
$t_s = 15$ s	$h_1 = 14$ cm
$q_1 = 16.6$ ml/s	$h_2 = 3$ cm
$q_2 = 0.55$ ml/s	$W_{a4} = 4.32 \times 10^{-4}$ M
$q_3 = 15.6$ ml/s	$W_{b4} = 5.28 \times 10^{-4}$ M
$q_{1e} = 16.6$ ml/s	$pH = 7$
$C_{v1} = 1.767$	$C_{v4} = 1.767$

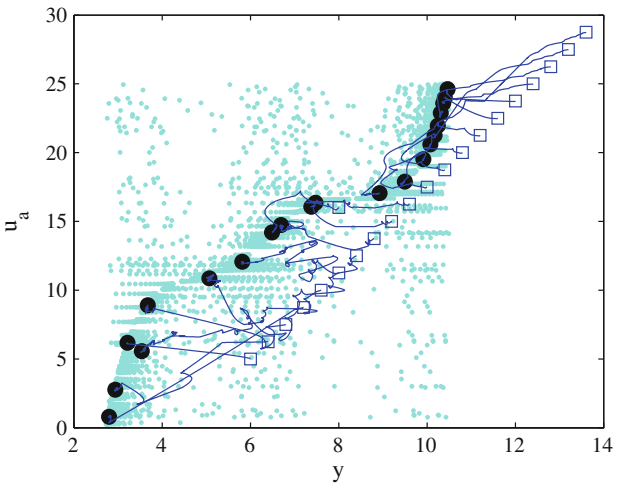

Fig. 8 The input–output sample distribution before the disturbance. The *small dots* present the samples, *big dots* the end cluster position and the *rectangles* the starting positions

Also after the model adapts its self to new process dynamics the control is better (Fig. 12).

The most important is that the error between the model and process output is smaller wit the adaptive control (Fig. 13).

The control with the adaptive FPFC is better than with the non-adaptive. There is less overshoot and better regulation after the disturbance. In Table 2 the sum squared errors are given. It can be seen that the error between model prediction and real process output is substantially smaller with the adaptive FPFC.

6 Mackey–Glass (M–G) time series prediction

To compare the proposed method to others, the results for the benchmark problem of 85-steps ahead Mackey-Glass

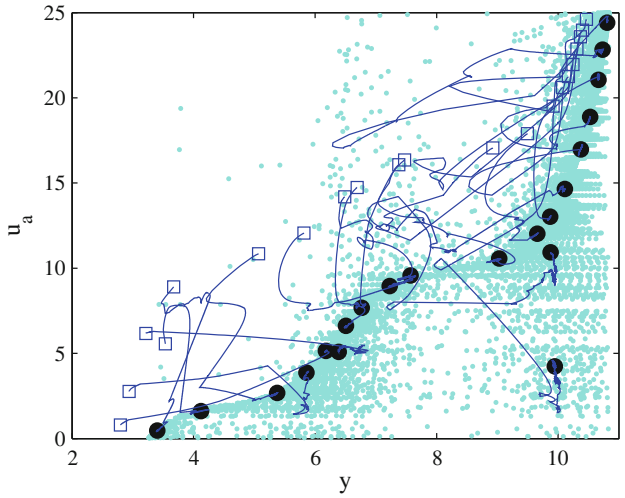

Fig. 9 The input–output sample distribution after the disturbance. The *small dots*present the samples, *big dots* the end cluster position and the *rectangles* the starting positions

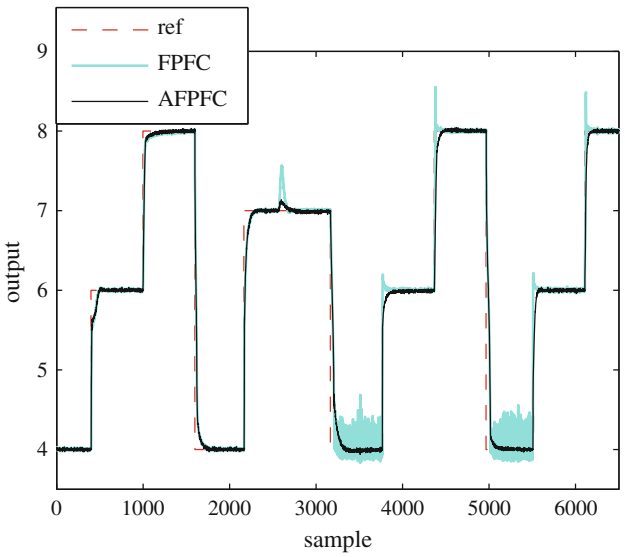

Fig. 10 Control of the process

time series (Mackey and Glass 1977) prediction are given. The detailed description of the experiment is given in (Kasabov and Song 2002). For the validation of the model we used the non-dimensional error index (NDEI), defined as the ratio of the root mean square error to the standard deviation of the target data. The results from other methods are taken from (Kasabov and Song 2002) and (Angelov 2010) and are shown in Table 3.

Some of the methods are also tested on the 6-step prediction. Therefore an experiment was also made for 6-step prediction. The proposed rGK approximated the 6-step prediction experiment [described in (Paiva and Dourado 2001)] with NDEI 0.0859 (10 clusters).

Table 2 The sum squared error

	APFC	FPFC
Before disturbance		
Model matching	86.9	90.9
Reference following	277.5	282.3
After disturbance		
Model matching	230.7	3.4e4
Reference following	784.1	750.9

Table 3 Comparison of recursive *GK* method and other fuzzy identification methods on a 85 steps ahead prediction of the M–G time-series

Methods	Rules (nodes, clusters)	NDEI
DENFIS	58	0.276
DENFIS	27	0.404
exTs	10	0.331
exTs	9	0.361
eTs	9	0.372
eTS	113	0.0954
eTs+	8	0.438
eTs+	10	0.392
RAN	113	0.375
ESOM	114	0.32
EFuNN	193	0.401
DENFIS	883	0.033
ESOM	1,000	0.044
Neural gas	1,000	0.062
EFuNN	1,125	0.094
rGK	3	0.4667
rGK	10	0.3787
rGK	58	0.2113
rGK	100	0.1166

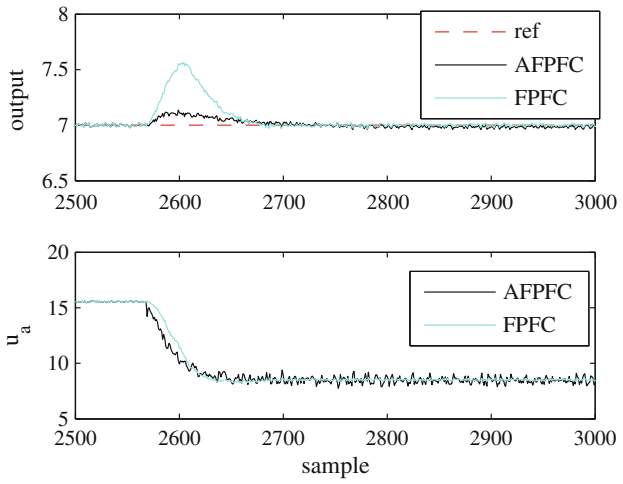

Fig. 11 Disturbance rejection

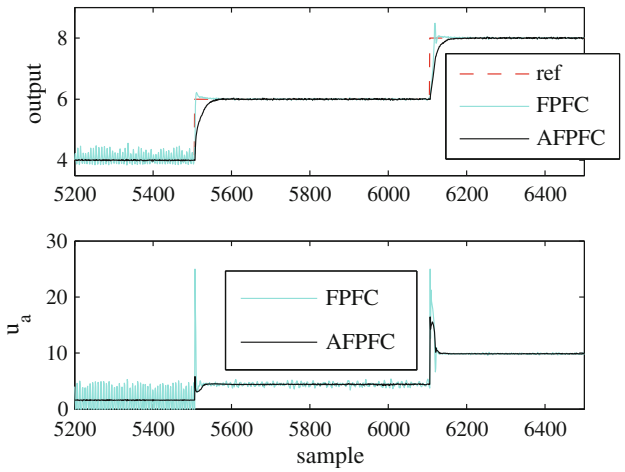

Fig. 12 Control after the disturbance

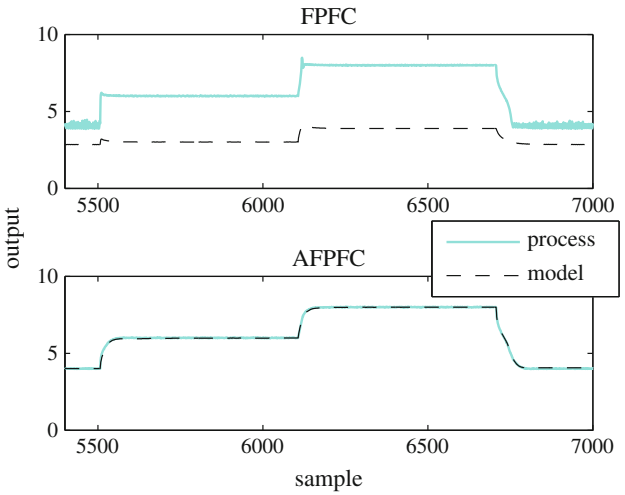

Fig. 13 Model matching after the disturbance

7 Conclusion

In this paper a method for on-line fuzzy model identification was presented. The method is based on a recursive Gustafson–Kessel algorithm and recursive fuzzy least squares. The rGK algorithm was derived from an off-line GK algorithm. Guidelines for parameters settings are given and examples of its use. The results for M–G time series prediction are given.

The proposed method can be used for adaptive fuzzy control, fault detection, model based design of experiments and other areas, where fuzzy models with time invariant parameters are inappropriate or where recursive identification should be used. The performance of the proposed method is comparable to other established on-line methods.

References

Angelov P (2004) An approach for fuzzy rule-base adaptation using on-line clustering. Integration of Methods and Hybrid Systems 35(3):275–289

Angelov P (2010) Evolving Takagi–Sugeno fuzzy systems from streaming data (eTs+). In: Angelov P, Filev D, Kasabov A (eds) Evolving intelligent systems: methodology and applications, Willey, IEE Press Series on Computational Intellegence, pp 273–300

Angelov P, Zhou X (2006) Evolving Fuzzy Systems from Data Streams in Real-Time, 2006 International Symposium on Evolving Fuzzy Systems, 7–9 September, 2006, Ambelside, Lake District, UK, IEEE Press, pp 29–35

Angelov PP (2002) Evolving rule-based models: a tool for design of flexible adaptive systems. Springer-Verlag, Heidelberg

Angelov PP, Filev DP (2004) An approach to online identification of Takagi–Sugeno fuzzy models. IEEE Trans Syst Man Cyber part B 34(1):484–497

Åström KJ, Wittenmark B (1995) Adaptive control. Addison-Wesley, New York

Azeem MF, Manmandlu M, Ahmad N (1999) Modified mountain clustering in dynamic fuzzy modeling, 2nd Int. Conf. Inform. Technol, Bhubaneswar, pp 63–68

Azeem MF, Hanmandlu M, Ahmad N (2003) Structure identification of generalized adaptive neuro-fuzzy inference systems. IEEE Trans on Fuzzy Syst 11(5):666–681

Bezdek JC (1981) Pattern recognition with fuzzy objective function algorithms. Plenum Press, New York

Chiu SL (1994) Fuzzy model identification based on cluster estimation. J Intell Fuzzy Syst 2:267–278

Dovžan D, Škrjanc I (2010) Fuzzy predictive functional control with adaptive fuzzy model. In: IEEE International Joint Conferences on Computational Cybernetics and Technical Informatics (ICCC-CONTI 2010), Timisoara, Romania, pp 143–147, 27–29 May 2010

Filev D, Georgieva O (2010) An extended version of the Gustafson–Kessel algorithm for evolving data stream clustering. In: Angelov P, Filev D, Kasabov A (eds) Evolving Intelligent Systems: Methodology and Applications. Willey, IEE Press Series on Computational Intellegence, pp 273–300

Goodwin GC, Sin KS (1984) Adaptive filtering prediction and control. Prentice-Hall, Upper Saddle River, New York

Gustafson D, Kessel W (1979) Fuzzy clustering with fuzzy covariance matrix, In: Proceedings of IEEE CDC, San Diego, CA, USA, pp 761–766

Hai-Jun Rong, Sundararajan N, Guang-Bin Huang, Saratchandran P (2006) Adaptive fuzzy inference system (SAFIS) for nonlinear system identification and prediction. Fuzzy Sets Syst 157(9):1260–1275

Henson MA, Seborg DE (1994) Nonlinear control of a pH neutralization process. IEEE Trans Control Syst Technol 2(3):169–182

Hwang CL, Chang LJ (2007) Neural-based control for nonlinear time-varying delay systems. IEEE Trans Syst Man Cyber part B 37(6):1471–1485

Johanson TA, Murray-Smith R (1997) The operating regime approach to nonlinear modeling and control. In: Murray-Smith R, Johanson TA (eds) Multiple model approaches to modeling and control. Taylor & Francis, London, pp 3–72

Juang CF, Lin CT (1998) An on-line self-constructing neural fuzzy inference network and its applications. IEEE Trans on Fuzzy Syst 6(1):12–32

Kasabov N (1996) Learning fuzzy rules and approximate reasoning in fuzzy neural networks and hybrid systems. Fuzzy Sets Syst 82(2):665–685

Kasabov N (1998a) ECOS: a framework for evolving connectionist systems and the ECO learning paradigm. In: Proceedings of ICONIP 1998, Japan, pp 1222–1235

Kasabov N (1998b) Evolving fuzzy neural networks-algorithms, applications and biological motivation. Methodologies for the conceptation. Design and Application of Soft Computing, Singapore, pp 271–274

Kasabov N (2001) Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning. IEEE Trans Syst Man Cyber part B 31(6):902–918

Kasabov NK, Song Q (2002) Dynamic evolving neural-fuzzy inference system and its application for time-series prediction. IEEE Trans on Fuzzy Syst 10(2):144–154

Kim E, Minkee Park, Ji S, Migon Park (1997) A new approach to fuzzy modeling. IEEE Trans on Fuzzy Syst 5(3): 328–337

Kim K, Baek J, Kim E, Park M (2005) TSK Fuzzy model based on-line identification. In: Proceedings of 11th IFSA World Congress, Beijing, China, pp 1435–1439

Kukolj D, Levi E (2004) Identification of complex systems based on neural and Takagi Sugeno fuzzy model. IEEE Trans Syst Man Cyber part B 34(1):272–282

Leng G, Prasad G, McGinnty TM (2004) On-line algorithm for creating self-organizing fuzzy neural networks. Neural Networks 17(10):1477–1493

Lin CT (1995) A neural fuzzy control system with structure and parameter learning. Fuzzy Sets Syst 70:183–212

Lin F-J, Lin C-H, Shen P-H (2001) Self-constructing fuzzy neural network speed controller for permanent-magnet synchronous motor drive. IEEE Trans Fuzzy Syst 9(5):751–759

Mackey MC, Glass L (1977) Oscilations and chaos in physiological control systems. Sci Agric 197:287–289

Paiva RP, Dourado A (2001) Structure and parameter learning of neuro-fuzzy systems: a methodology and a comparative study. J Intell Fuzzy Syst 11:147–161

Qiao J, Wang H (2008) A self-organizing fuzzy neural network and its applications to function approximation and forecast modeling. Neurocomputing 71(4–6):564–569

Shing J, Jang R (1993) ANFIS: adaptive-network-based fuzzy inference system. IEEE Trans Syst Man Cyber 23(3):665–685

Škrjanc I, Matko D (2000) Predictive functional control based on fuzzy model for heat-exchanger pilot plant. IEEE Trans Fuzzy Syst 8(6):705–712

Soleimani-B H, Lucas C, Araabi BN (2010) Recursive Gath-Geva clustering as a basis for evolving neuro-fuzzy modeling, Evolving Systems, Springer 1(1):59–71. doi:10.1007/s12530-010-9006-x

Takagi T, Sugeno M (1985) Fuzzy identification of systems and its applications. IEEE Trans Syst Man Cyber, vol SMC-15, 116–132

Tzafestas SG, Zikidis KC (2001) NeuroFAST: On-Line Neuro-Fuzzy ART-Based Structure and Parameter Learning TSK Model. IEEE Trans Syst Man Cyber part B 31(5):797–802

Wang D, Xiao-Jun Zeng, Keane JA (2008) An incremental construction learning algorithm for identification of T–S Fuzzy Systems. In: Proceedings of FUZZ 2008 (IEEE International Conference on Fuzzy Systems 2008), Hong Kong, pp 1660–1666

Wu S, Er MJ (2000) Dynamic fuzzy neural networksa novel approach to function approximation. IEEE Trans Syst Man Cyber part B 30(2):358–364

Wu S, Er MJ, Gao Y (2001) A fast approach for automatic generation of fuzzy rules by generalized dynamicc fuzzy neural networks. IEEE Trans on Fuzzy Syst 9(4):578–594

Xu L, Krzyzak A, Oja E (1993) Penalized competitive learning for clustering analysis, RBF net, and curve detection. IEEE Trans Neural Networks 4:636–649

Yager RR, Filev DP (1993) Learning of fuzzy rules by mountain clustering. Proc SPIE Conf Applicat Fuzzy Logic Technol, Boston MA, pp 246–254