Robust Kernel Clustering Algorithm for Nonlinear System Identification

Mohamed Bouzbida, Lassad Hassine, and Abdelkader Chaari

1National Higher Engineering School of Tunis (ENSIT), University of Tunis, 5 Av. Taha Husein, BP 56, 1008 Tunis, Tunisia
2Laboratoire d’Ingenierie des Systemes Industriels et des Energies Renouvelables (LISIER), University of Tunis, ENSIT, Tunis, Tunisia

Correspondence should be addressed to Mohamed Bouzbida; bouzbida_mohamed@hotmail.fr

Received 12 December 2016; Revised 16 March 2017; Accepted 30 March 2017; Published 14 May 2017

Copyright © 2017 Mohamed Bouzbida et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In engineering field, it is necessary to know the model of the real nonlinear systems to ensure its control and supervision; in this context, fuzzy modeling and especially the Takagi-Sugeno fuzzy model has drawn the attention of several researchers in recent decades owing to their potential to approximate nonlinear behavior. To identify the parameters of Takagi-Sugeno fuzzy model several clustering algorithms are developed such as the Fuzzy $C$-Means (FCM) algorithm, Possibilistic $C$-Means (PCM) algorithm, and Possibilistic Fuzzy $C$-Means (PFCM) algorithm. This paper presents a new clustering algorithm for Takagi-Sugeno fuzzy model identification. Our proposed algorithm called Robust Kernel Possibilistic Fuzzy $C$-Means (RKPFCM) algorithm is an extension of the PFCM algorithm based on kernel method, where the Euclidean distance used the robust hyper tangent kernel function. The proposed algorithm can solve the nonlinear separable problems found by FCM, PCM, and PFCM algorithms. Then an optimization method using the Particle Swarm Optimization (PSO) method combined with the RKPFCM algorithm is presented to overcome the convergence to a local minimum of the objective function. Finally, validation results of examples are given to demonstrate the effectiveness, practicality, and robustness of our proposed algorithm in stochastic environment.

1. Introduction

Modeling and identification are significant steps in the design of the control system. Typical applications of these models are the simulation, the prediction, or the control system design. Generally, the modeling process consists of obtaining a parametric model with the same dynamic behavior of the real process. However, when the process is nonlinear and complex, it is very difficult to define the different mathematical or physical laws which describe its behavior [1, 2]. In this context, the modeling of nonlinear systems by the conventional methods is very difficult and occasionally ineffective [3, 4]. So, other nonconventional techniques based on fuzzy logic are used more often in modeling this kind of process due to excellent ability of describing its behavior [4–6].

Among the best fuzzy modeling approaches developed in literature we mention the Takagi-Sugeno fuzzy model. In effect, this model is described by if-then rules. Each rule includes a fuzzy set antecedent and mathematical functions as consequent representing the process behavior in each region [2, 3, 7]. The identification problem consists of estimating the model parameters. In this context, to identify the parameters of Takagi-Sugeno fuzzy model, many techniques were developed such as the Adaptive schemes, heuristic approaches, nearest neighbor clustering, and support vector learning mechanisms. Besides fuzzy clustering algorithms are widely used in fuzzy modeling. Fuzzy $C$-Means (FCM), Gustafson-Kessel (G-K), Gath-Geva (G-G), Possibilistic $c$-Means (PCM), Fuzzy C-Regression Model (FCRM), Enhanced Fuzzy C-Regression Model (EFCRM), and Possibilistic Fuzzy $c$-Means (PFCM) are popular clustering algorithms used in structure identification part and LS, WRLS, and orthogonal least square (OLS) technique were applied in consequent parameter estimation. Among the clustering algorithms, Fuzzy $c$-Means (FCM) developed
by Bezdek is well-known but this algorithm is sensitive
to noise or outliers and susceptible to local minima [8, 9]. However, noise in the data sets can make the situa-
tion worse by creating many inauthentic minima. These
are able to distort the global minimum solution found by
FCM algorithm. This flaw has stimulated the researchers
to overcome this inconvenience. To fight against the effects
of outlying data, various approaches are considered such as the
possibilistic clustering (PCM) proposed by Krishnapuram
and Keller [10] and fuzzy noise clustering approach of
Dave [11]. The possibilistic approach executed a possibilistic
partition, in which a membership relation calculates the
absolute degree of typicality of a point in a cluster. Although
the PCM algorithm is robust against the noise points and
allows identifying these outliers, it is very responsive to
initializations and occasionally generates coincident clusters.

To solve this deficiency of identical clusters, a Possibilistic
Fuzzy c-Means (PFCM) algorithm was suggested by Wu
and Zhou in 2006. Nonetheless, these algorithms are not
efficient for unequal dimension clusters and cannot separate
clusters that are nonlinearly separable in input space and
their limits between two clusters are linear. In our work to
overcome this shortcoming, kernel methods [12] are regarded
as the way of dealing with this problem. We propose a new
clustering algorithm called Robust Kernel Possibilistic Fuzzy
C-Means (RKPFCM), which adopts a kernel induced metric
in the data space to replace the original Euclidean norm
metric. By changing the inner product with an appropriate
hyper tangent kernel function, one can implicitly affect a
nonlinear mapping to a high dimensional feature space where
the data is more clearly separable. However, our proposed
algorithm RKPFCM has an iterative nature that makes it
sensitive to initialization and sensitive to converge to a local
minimum. To overcome these problems, several solutions
have been proposed in the literature. Among them is com-
bining the clustering algorithm with a heuristic optimization
technique. In this context, many researches have proposed
the evolutionary computation technique based on Particle
Swarm Optimization (PSO). They have been successfully
applied to solve various optimization problems [13]. Thus,
we introduce PSO into the RKPFCM algorithm to achieve
global optimization. The efficacy of our algorithm compared
to many algorithms is tested on noisy nonlinear systems
defined by recurrent equations and an application to Box
Jenkins system. This paper will be presented as follows.

The second part of work is reserved for introducing the
Takagi-Sugeno fuzzy model. The third part will be devoted
to identifying the premise parameters of this model where
we used the proposed RKPFCM algorithm and PSO algorithm
is introduced. In the fourth part, we will focus on identification
of consequent parameters. The simulations results and the
model validity of RKPFCM and RKPFCM-PSO are presented
in part five. Finally, we conclude this paper.

2. Takagi-Sugeno Fuzzy Model

Takagi-Sugeno fuzzy model (T-S) is one of the best tech-
niques used for modeling a nonlinear system represented
by the recurrent equation \( y(k) = \beta_T x_k \). T-S model is
constructed by a rule-based type “if...then” in which the
consequent uses numeric variables rather than linguistic vari-
ables (case of Mandani). The consequent can be expressed by
a constant, a polynomial, or differential equation depending
on the antecedent variables. The T-S fuzzy model allows
approximating the nonlinear system into several locally linear
subsystems [4, 17].

In general, a Takagi-Sugeno fuzzy model is based on if...
then rules of the form

\[
R_i : \text{if } x_{ik} \text{ is } A_{i1}, \ldots, x_{nk} \text{ is } A_{in} \text{ then } y_i = a_i^T x_k + b_i. \tag{1}
\]

The “if” rule function defines the premise part and “then”
rule function constitutes the consequent part of the T-S fuzzy
model.

\[
R_i: \text{represents the } i\text{th rule;}
\]

\[
a_i = [a_{i1}, a_{i2}, \ldots, a_{in}] : \text{is the parameters vector, such as } a_i \in \mathbb{R}^n,
\]

\[
b_i : \text{is a scalar;}
\]

\[
x_k = [x_{1k}, x_{2k}, \ldots, x_{nk}]^T : \text{is observations vector;}
\]

\[
A_{i1}, A_{i2}, \ldots, A_{in} : \text{represents the fuzzy subsets,}
\]

where \( i \in [1, \ldots, C] \).

Here, the fuzzy sets are represented by the following
membership function [7]

\[
A_{ij}(x_{jk}) = \exp \left( -\frac{(x_{jk} - v_{ij})^2}{\sigma_{ij}^2} \right) \in [0, 1]
\]

\[
(i = 1, 2, \ldots, C, \ j = 1, 2, \ldots, n),
\]

where \( v_i = [v_{i1}, v_{i2}, \ldots, v_{in}]^T \in \mathbb{R}^n \) are centers and \( \sigma_i = [\sigma_{i1}, \sigma_{i2}, \ldots, \sigma_{in}]^T \in \mathbb{R}^n \) is the width of the membership function.

The estimated output model is defined by the following
equation [4]:

\[
y(k) = \sum_{i=1}^{C} \beta_i(k) y_i(k). \tag{3}
\]

As

\[
\beta_i(k) = \frac{\prod_{j=1}^{n} A_{ij}(x_{jk})}{\sum_{C} \prod_{j=1}^{n} A_{ij}(x_{jk}) } k = 1, 2, \ldots, N, \tag{4}
\]

so,

\[
\bar{y}(k) = \sum_{i=1}^{C} \beta_i(k) \left[ a_i^T x_k + b_i \right]. \tag{5}
\]

3. Identification Algorithm for
Premise Parameters

To identify the premise parameters of a Takagi-Sugeno fuzzy
model described by equation (1), we used the Possibilistic
Fuzzy C-Means (PFCM) algorithm and our proposed algo-
rithms (RKPFCM and RKPFCM-PSO).
3.1. Possibilistic Fuzzy C-Means (PFCM) Algorithm. The Possibilistic Fuzzy C-Means (PFCM) algorithm, which uses Euclidean distance, finds the partition of the collection $X = \{x_1, \ldots, x_N\} \subset \mathbb{R}^D$ of $N$ measures, specified by $k$-dimensional vectors $x_i = [x_{i1}, x_{i2}, \ldots, x_{iD}]^T$, into $C$ fuzzy subsets by minimizing the following objective function [18]:

$$J_{\text{PFCM}}(U, T, V) = \sum_{i=1}^{C} \sum_{k=1}^{N} (a \mu_{ik}^m + b t_{ik}) D_{ik}^2$$

(6)

$$+ \sum_{i=1}^{C} \sum_{k=1}^{N} (t_{ik} \log (t_{ik}) - t_{ik}),$$

where

$$1 \leq C \leq N : \text{the number of clusters;}$$

$$\mu_{ik} : \text{the membership of } x_k \text{ in cluster } i \text{ satisfying}$$

$$0 \leq \mu_{ik} \leq 1,$$

$$\sum_{i=1}^{C} \mu_{ik} = 1 \quad 1 \leq k \leq N;$$

$$t_{ik} : \text{the typicality of } x_k \text{ in classes } i;$$

$$D_{ik}^2 = \|x_k - v_i\|^2;$$

(8)

$$V : \text{the set of cluster centers } (v_i \in \mathbb{R}^D);$$

$$\eta_i : \text{the suitable positive numbers described by}$$

$$\eta_i = K \frac{\sum_{k=1}^{N} \mu_{ik}^m V_{ik}^2}{\sum_{k=1}^{N} \mu_{ik}^m}, \quad K > 0.$$  

(9)

Typically, $K$ is chosen to be 1.

$\mu_{ik,\text{FCM}}$ are the terminal membership values of FCM.

$m$ is a weighting degree; this parameter has a significant impact on the form of clusters in data space.

To minimize equation (6), we take its partial derivative of variables, $\mu_{ik}, t_{ik},$ and $v_i,$ equal to zero and obtain the following equations:

$$\mu_{ik} = \left[ \sum_{j=1}^{C} \left( \frac{D_{jk}}{D_{ik}} \right)^{2/(m-1)} \right]^{-1},$$

(10)

$$t_{ik} = \exp \left( - \frac{b D_{ik}^2}{\eta_i} \right), \quad \forall i, k,$$

(11)

$$v_i = \frac{\sum_{k=1}^{N} (a \mu_{ik}^m + b t_{ik}) x_k}{\sum_{k=1}^{N} (a \mu_{ik}^m + b t_{ik})}, \quad \forall i.$$  

(12)

**PFCM Algorithm Steps.** Given a set of observations $X = [x_1, \ldots, x_N]^T.$

**Initialization** ($l = 0$)

Set the number of clusters $C, 1 \leq C \leq N.$

Set the level of weighting $m: 2 \leq m < 4.$

Set the parameters $a, b.$

Set the stopping criterion $\varepsilon: \varepsilon > 0.$

Execute a FCM clustering algorithm to find initial fuzzy partition matrix $U$ and cluster centers $V.$

Initialize the typicality matrix $T$ randomly.

Compute $\eta_i$ by (9).

Repeat for $l = 1, 2, \ldots$.

**Step 1.** Compute the cluster centers by (12).

**Step 2.** Compute the membership matrix $U = [\mu_{ik}]$ by (10).

**Step 3.** Compute the typicality matrix $T = [t_{ik}]$ by (11).

Until $\|U^l - U^{l-1}\| < \varepsilon;$ then stop. Otherwise, set $l = l + 1$ and return to Step 1.

3.2. Proposed Robust Kernel Possibilistic Fuzzy C-Means (RKPFCM) Algorithm. The PFCM can deal with noisy data better than FCM and PCM; nevertheless, these conventional clustering algorithms become more effective when applied on linearly separable data or with a reasonable quantity of errors. In reality, the linearly separable data are rare. Therefore, FCM, PCM, and PFCM share the same negative point in that they are unable to get good separation of data that are nonlinearly separable in input space. To correct the imperfections found in PFCM particularly the nonlinear separable problem, kernel [7] methods are regarded as the way of dealing with this problem. In this context, we proposed a new extension of Possibilistic Fuzzy C-Means algorithm based on kernel method (RKPFCM). The present work proposes a way of increasing the accuracy of the PFCM algorithm by exploiting hyper tangent kernel function to calculate the distance used in its objective function.

The kernel function is defined as a generalization of the distance metric that measures the distance between two data points mapped into a future space in which the data are more clearly separable [12, 19–21].

Define a nonlinear map as $\Phi : x \rightarrow \Phi(x) \in F,$ where $x \in X,$ and $F$ is the transformed feature space with higher or even infinite dimension. $X$ denotes the data space mapped into $F$ [20–22].

The RKPFCM algorithm minimizes the following objective function:

$$J_{\text{RKPFCM}}(U, T, V)$$

$$= \sum_{i=1}^{C} \sum_{k=1}^{N} (a \mu_{ik}^m + b t_{ik}) \left\| \phi (x_k) - \phi (v_i) \right\|^2$$

(13)

$$+ \sum_{i=1}^{C} \eta_i \sum_{k=1}^{N} (t_{ik} \log (t_{ik}) - t_{ik}).$$
Then $\|x_k - v_j\|$ is mapped into space $F$ [22]:

$$\|x_k - v_j\| \rightarrow \|\phi(x_k) - \phi(v_j)\|,$$

where

$\|\phi(x_k) - \phi(v_j)\|^2 = (\phi(x_k) - \phi(v_j)) \cdot (\phi(x_k) - \phi(v_j))$

$$= \phi(x_k) \cdot \phi(x_k) + \phi(v_j) \cdot \phi(v_j) - 2\phi(x_k) \cdot \phi(v_j)$$

$$= K(x_k, x_k) + K(v_j, v_j) - 2K(k_k, v_j).$$

$K(x_k, v_j) = \Phi(x_k)^T \Phi(v_j)$ is an inner product kernel function. If we adopt the hyper tangent kernel function, that is,

$$K(x_k, v_j) = 1 - \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right) \quad \text{where} \quad \sigma > 0, \quad (16)$$

then $K(x_k, x_k) = K(v_j, v_j) = 1$. Thus (15) can be written as

$$\|\phi(x_k) - \phi(v_j)\|^2 = 2 \left(1 - K(x_k, v_j)\right)$$

$$= 2 \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right). \quad (17)$$

Considering (17), the objective function (13) is transformed as follows:

$$J_{RKPFCM}(U, T, V)$$

$$= 2 \sum_{i=1}^{C} \sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik}) \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)$$

$$+ \sum_{i=1}^{C} \eta_i \sum_{k=1}^{N} (t_{ik} \log(t_{ik}) - t_{ik}). \quad (18)$$

The derivation of the objective function (18) according to $\mu_{ik}$, $t_{ik}$, and $v_j$, defines the relationship update of cluster centers and membership coefficients.

(i) Derivative of $J_{RKPFCM}(U, T, V)$ with respect to $v_j$.

$$\frac{\partial J_{RKPFCM}(U, T, V)}{\partial v_j} = 2 \sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik})$$

$$\cdot \left(1 - \tanh^2\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)\right) \frac{x_k - v_j}{\sigma^2}$$

$$= 2 \sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik}) \left(1 - \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)\right)$$

$$\cdot \left(1 + \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)\right) \frac{x_k - v_j}{\sigma^2}. \quad (19)$$

So,

$$\frac{\partial J_{RKPFCM}(U, T, V)}{\partial v_j} = 2 \sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik}) K(x_k, v_j)$$

$$\cdot \left(1 + \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)\right) \frac{x_k - v_j}{\sigma^2}. \quad (20)$$

Equating (20) to zero leads to

$$\frac{\partial J_{RKPFCM}(U, T, V)}{\partial v_j} = 0. \quad (21)$$

Then,

$$v_j = \frac{\sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik}) K(x_k, v_j) \left(1 + \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)\right)x_k}{\sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik}) K(x_k, v_j) \left(1 + \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)\right)} \quad (22)$$

(ii) Derivative of $J_{RKPFCM}(U, T, V)$ with respect to $\mu_{ik}$. In this part we used the Lagrange multiplier

$$J_{RKPFCM}(U, P, V)$$

$$= 2 \sum_{i=1}^{C} \sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik}) \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)$$

$$+ \sum_{i=1}^{C} \eta_i \sum_{k=1}^{N} (t_{ik} \log(t_{ik}) - t_{ik})$$

$$- \sum_{k=1}^{N} \left(\lambda_k \cdot \left(\sum_{i=1}^{C} \mu_{ik} - 1\right)\right), \quad (23)$$

$$\frac{\partial J_{RKPFCM}(U, T, V)}{\partial \lambda_i} = - \left(\sum_{k=1}^{C} \mu_{ik} - 1\right) = 0, \quad (24)$$

$$\frac{\partial J_{RKPFCM}(U, T, V)}{\partial \mu_{ik}} = 2 \sum_{i=1}^{C} \sum_{k=1}^{N} (a\mu_{ik}^m + bt_{ik}) \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)$$

$$- \lambda = 0. \quad (25)$$

From expression (25), we can write $\mu_{ik}$ in this form:

$$\mu_{ik} = \left(\frac{\lambda}{2ma \tanh\left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)}\right)^{1/(m-1)} \quad (26)$$
Substituting expression (26) in expression (24):

$$
\frac{\partial J_{RKPFCM}(U, T, V)}{\partial \lambda} = \sum_{j=1}^{C} \mu_{jk} \frac{\lambda}{2ma \tanh \left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)} \left(\frac{1}{m-1}\right)
$$

$$
= \sum_{j=1}^{C} \frac{1}{1/(m-1)} = 1. \tag{27}
$$

It is also

$$
\left(\frac{\lambda_k}{2ma}\right)^{1/(m-1)} = \frac{1}{\sum_{j=1}^{C} \frac{1}{1/(m-1)}}. \tag{28}
$$

The two expressions (26) and (28) give the following expression:

$$
\mu_{ik} = \frac{1}{\sum_{j=1}^{C} \left(\frac{1}{1/(m-1)} \tanh \left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right) / \tanh \left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)\right)^{1/(m-1)}}. \tag{29}
$$

Therefore, the updating relationship is

$$
\mu_{ik} = \left[\sum_{j=1}^{C} \frac{1}{1/(m-1)} \tanh \left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right) / \tanh \left(-\frac{\|x_k - v_j\|^2}{\sigma^2}\right)\right]^{-1}. \tag{30}
$$

(iii) Derivative of $J_{RKPFCM}(U, T, V)$ with respect to $t_{ik}$.

$$
\frac{\partial J_{RKPFCM}(U, T, V)}{\partial t_{ik}} = 2b \tanh \left(-\frac{\|x_k - v_i\|^2}{\sigma^2}\right) + \eta_i \log t_{ik} = 0, \tag{31}
$$

$$
t_{ik} = \exp \left(-\frac{2b \left(1 - K(x_k, v_i)\right)}{\eta_i}\right), \quad \forall i, k. \tag{32}
$$

Therefore, the updating typicality matrix is

$$
t_{ik} = \exp \left(-\frac{2b \left(1 - K(x_k, v_i)\right)}{\eta_i}\right), \quad \forall i, k. \tag{33}
$$

Similarly (9) is rewritten by

$$
\eta_i = K \frac{\sum_{k=1}^{N} \mu_{ik}^{m_{FCM}} \left(1 - K(x_k, v_i)\right)}{\sum_{k=1}^{N} \mu_{ik}^{m_{FCM}}}. \tag{34}
$$

RKPFCM Algorithm Steps. Given a set of observations $X = [x_1, \ldots, x_N]^T$.

Initialization ($l = 0$)

Set the number of clusters $C$, $1 \leq C \leq N$.

Set the level of weighting $m$: $2 \leq m < 4$.

Set the parameters $a, b, \text{ and } \sigma$.

Set the stopping criterion $\varepsilon$: $\varepsilon > 0$.

Execute a FCM clustering algorithm to find initial fuzzy partition matrix $U$ and cluster centers $V$.

Initialize the typicality matrix $T$ randomly.

Compute $\eta_i$ by (34).

Repeat for $l = 1, 2, \ldots$

Step 1. Compute the cluster centers by (22).

Step 2. Compute the membership matrix $U = [\mu_{ik}]$ by (30).

Step 3. Compute the typicality matrix $T = [t_{ik}]$ by (33).

Until $\|U^l - U^{l-1}\| < \varepsilon$; then stop. Otherwise, set $l = l + 1$ and return to Step 1.

3.3. Robust Kernel Possibilistic Fuzzy C-Means Algorithm Based on PSO (RKPFCM-PSO)

3.3.1. PSO. The Particle Swarm Optimization is a heuristic search method proposed by Kennedy and Eberhart (1995). This technique uses random population solution particles to find an optimal solution to problems. Each particle moves in the search space with a dynamically adjusted position and velocity for the best solution. The particle is characterized by data structure that contains the coordinates of the current position in the search space, the best solution point visited so far, and the subset of other agents that are seen as neighbors. These adjustments are based on the historical behaviors of itself and other agents in the swarm. The change of speed (acceleration) and the position of each particle in the optimization landscape (search space) are iteratively [6, 12, 23]

$$
v_{ik}^{(l+1)} = v_{ik}^{(l)} + \rho_1 \left(p_k^{(l)} - x_{ik}^{(l)}\right) + \rho_2 \left(p_g^{(l)} - x_{ik}^{(l)}\right), \tag{35}
$$

$$
x_{ik}^{(l+1)} = x_{ik}^{(l)} + v_{ik}^{(l+1)},
$$

where $p_k^{(l)}$ is the personal best solution of particle $k$, and $p_g^{(l)}$ is the global best solution of the swarm up to iteration $l$. The constants $\rho_1$ and $\rho_2$ are the acceleration coefficients, and $\eta_1$ and $\eta_2$ are the parameters that regulate the balance between the local and global search.
where

\[ k = 1, \ldots, NP: \text{size of particles}; \]
\[ D: \text{size of the landscape (search space)}; \]
\[ v_k = (v_{k1}, \ldots, v_{kd}, \ldots, v_{kD}): \text{the speed of particle}; \]
\[ x_k = (x_{k1}, x_{k1}, \ldots, x_{kd}, \ldots, x_{kD}): \text{the position of particle}; \]
\[ p_k = (p_{k1}, p_{k1}, \ldots, p_{kd}, \ldots, p_{kD}): \text{the best previous position of particle}; \]
\[ g: \text{index represents the best particle among all particles in the group}; \]
\[ K: \text{the constriction factor described by the following relationship:} \]
\[ K = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}}, \quad \text{where} \varphi = c_1 + c_2, \quad (36) \]

where \( c_1 \) and \( c_2 \) are two positive constants satisfying the following relationship:
\[ \varphi = c_1 + c_2 > 4; \quad (37) \]
\[ \rho_1 \text{ and } \rho_2: \text{random variables defined as follows:} \]
\[ \rho_1 = r_1 \times c_1, \]
\[ \rho_2 = r_2 \times c_2, \quad (38) \]

where \( r_1 \) and \( r_2 \) are two random variables between 0 and 1;
\[ w: \text{the weight of inertia according to this equation:} \]
\[ w = w_{\text{max}} - \frac{w_{\text{max}} - w_{\text{min}}}{\text{iter}_{\text{max}}} \times \text{iter}, \quad (39) \]

where \( w_{\text{max}} \) and \( w_{\text{min}} \) are the initial and final weight, \( \text{iter}_{\text{max}} \) is the maximum iterations, and \( \text{iter} \) is the current iteration number.

3.3.2. Fitness Function. The fitness function defines our optimization problem described by the following expression:
\[ \text{Fitness} = \frac{G}{J_{\text{RKPFCM}}(U, T, V)}, \quad (40) \]

where
\[ G \text{ is a positive constant.} \]
\[ J_{\text{RKPFCM}}(U, T, V) \text{ represents the objective function of the RKPFCM algorithm.} \]

RKPFCM-PSO Algorithm. Given a set of observations \( X = [x_1, x_2, \ldots, x_N]^T \), the RKPFCM-PSO algorithm is described by the following steps.

Initialization \((l = 0)\)

Select the number of clusters \( C \), fuzzy degree \( m \), the parameters \( a \) and \( b \), the population size \( NP \), the constants \( c_1 \) and \( c_2 \), the random variables \( r_1 \) and \( r_2 \), the weight of inertia \( w_{\text{max}} \) and \( w_{\text{min}} \), the size of the search space \( D \), the constant \( G \), and the stopping criterion \( \varepsilon \).

Set the 1st particle generation clusters centers.

Initialize the fitness function and speed of each particle.

Compute \( \eta_i \) by (34).

Repeat \( I = I + 1 \)

Step 1. Compute the fuzzy partition matrix \( U = [\mu_{ik}] \) by (30).

Step 2. Compute the typicality matrix \( T = [t_{ik}] \) by (33).

Step 3. Calculate the new value of fitness for each particle using (40).

Step 4. Compare the fitness of each particle with \( p_{\text{best}} \), if the value is better than \( p_{\text{best}} \) and then set the \( p_{\text{best}} \) value.

Step 5. Compare the fitness value of \( g_{\text{best}} \) with the following: if the value is better than \( g_{\text{best}} \), \( g_{\text{best}} \) then is set equal to this value.

Step 6. Update position and speed of each particle by (35).

So this algorithm is converged when \( \|v_k(\text{iter} + 1) - v_k(\text{iter})\| < \varepsilon \); that is to say, stop iteration and find the best solution in the last generation. If not, go back to Step 1.

4. Identification for Consequent Parameters

The defuzzification method, used in the Takagi-Sugeno fuzzy model, is linear with the consequent parameters \( \theta_i = [a_i^T, b_i] \) which can be obtained as a solution of a weighted least squares problem according to the following equation:
\[ \theta_i = \left[X_e^T \psi_i X_e \right]^{-1} X_e^T \psi_i Y, \quad (41) \]

where
\[ X_e = [X; 1] \text{ represents an extension of regression} \]
\[ X = [X_1^T, X_2^T, \ldots, X_N^T]^T; \]
\[ Y = [y_1, y_2, \ldots, y_N]^T \text{ is the output vector;} \]
\[ \psi_i \text{ is a diagonal matrix of dimension } (N \times N) \text{ containing the coefficients } \mu_{ik} \text{ of fuzzy memberships}. \]

The RKPFCM and RKPFCM-PSO clustering algorithm are used to find width of the membership functions by the following equation [24]:
\[ \sigma_{ij} = \sqrt{\frac{2 \times \left[ \sum_{k=1}^{N} \mu_{ik} (x_{jk} - v_{ij})^2 \right]}{\sum_{k=1}^{N} \mu_{ik}}}, \quad (42) \]
5. Simulation Results and Validation Model

5.1. Identical Data with Noise. In this example we have used 12 data which are composed of 10 models and two noises; this data set (X12) is presented in [25]. The FCM, PCM, PFCM, KPFCM, and our algorithms RKPFCM and RKPFCM-PSO were used in clustering the data set in two groups (C = 2).

In this example the parameters settings are \(a = 1, b = 2, m = 2, G = 10, NP = 20, D = 2, w_{\text{max}} = 0.9, w_{\text{min}} = 0.4, c_1 = 2.05, c_2 = 2.05, \sigma = 20, \text{iter}_{\text{max}} = 1000, \) and \(\epsilon = 10^{-9}\).

Figure 1 shows the clustering results for our proposed method RKPFCM-PSO. The ideal (true) centroids are

\[
V_{\text{ideal}} = \begin{bmatrix} -3.34 & 0 \\ 3.34 & 0 \end{bmatrix}.
\]

Table 1 shows the results of center clusters using the six algorithms. The error between the results prototypes and ideal center clusters is calculated by the next expression:

\[
E_* = \|V_{\text{ideal}} - V_*\|^2,
\]

where \(*\) is the FCM, PCM, PFCM, KPFCM, RKPFCM, and RKPFCM-PSO.

According to Table 1, our proposed algorithm RKPFCM-PSO gives the best prototypes of centers.

Figure 1 shows the effectiveness of our approach as well.

5.2. Identification of T-S Fuzzy Model. After applying the identification algorithm, it is necessary to validate the Takagi-Sugeno fuzzy model. Several validation tests of the model are used. Among them, we cited the Mean Square Error (MSE) test, Root Mean Square Error (RMSE), and the Variance Accounting For (VAF) test.

\[
\text{MSE} = \frac{1}{N} \sum_{k=1}^{N} (y(k) - y_{\text{est}}(k))^2,
\]

\[
\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} (y(k) - y_{\text{est}}(k))^2},
\]

\[
\text{VAF} = 100\% \left[ 1 - \frac{\text{var}(y - y_{\text{est}})}{\text{var}(y)} \right],
\]

where \(y\) is the real output and \(y_{\text{est}}\) is the estimated output.

5.2.1. Example 1. Consider a nonlinear system described by the following difference equation [15]:

\[
y(k) = y^2(k-3) + y^2(k-2) + y^2(k-1) + \tanh(u(k)) + \frac{1}{1 + y^2(k-1) + y^2(k-2) + y^2(k-3)} + e(k),
\]

where \(y(k)\) and \(u(k)\) are the output and the input of the system, respectively. \(e(k)\) is a noise.

\[
u(k) = 0.6 \sin(3\pi k T_s) + 0.2 \sin(4\pi k T_s) + 1.2 \sin(\pi k T_s).
\]

\(T_s = 0.01\).

200 samples were generated by simulation and were used, where the selected input variables are chosen \(\{y(k-1), y(k-2), y(k-3), u(k)\}\).

The complete data set has been used to train the model. The noise influence is analyzed with different SNR levels (SNR = 10 dB and SNR = 5 dB).

In this part, we have applied various algorithms and our proposed clustering RKPFCM and RKPFCM-PSO which approximate the nonlinear model (46).

The used parameters are \(a = 1, b = 3, m = 2, G = 10, NP = 30, D = 4, w_{\text{max}} = 0.9, w_{\text{min}} = 0.4, c_1 = 2.05, c_2 = 2.05, \sigma = 20, \text{iter}_{\text{max}} = 1000, \) and \(\epsilon = 10^{-9}\).

The shape of the excitation signal used for identification is illustrated in Figure 2.
The simulation result given by the RKPFCM-PSO algorithm is illustrated in Figure 3. Table 2 shows the various modeling performance results without noise obtained by different algorithms; this comparison results demonstrate that the best MSE and best VAF are obtained by the proposed methods (RKPFCM and RKPFCM-PSO).

Tables 3 and 4 present the various modeling performance results with noise influence (SNR = 5 dB and 10 dB) obtained by the different algorithms. However, our proposed algorithm RKPFCM-PSO retained the best performance with a higher level of noise.

The local linear models identified are given as follows:

\[ y_1(k) = 0.3167y(k - 1) + 0.0263y(k - 2) - 0.0482y(k - 3) + 0.1183(k) + 0.6916, \]

\[ y_2(k) = 0.4045y(k - 1) + 0.0665y(k - 2) - 0.0304y(k - 2) + 0.1519u(k) + 0.5417. \]
1500 samples were generated by simulation in which 1000 samples were used to train the model. Fuzzy model parameters have been identified once, testing of model was done by the remaining 500 samples, and \{y(k - 1), y(k - 2), u(k), u(k - 1)\} are chosen as input variables. In this example the parameters settings are \(a = 1, b = 2, m = 2, G = 10, N P = 30, D = 4, w_{\text{max}} = 0.9, w_{\text{min}} = 0.4, c_1 = 2, c_2 = 2.1, \sigma = 20, \text{iter}_{\max} = 1000, \) and \(\epsilon = 10^{-9}.\)

The noise influence is analyzed with different SNR levels (SNR = 20 dB, SNR = 10 dB, SNR = 5 dB, and SNR = 1 dB).

The obtained identification results by RKPFCM-PSO are, respectively, shown in Figures 4 and 5.

The evaluation performance index (RMSE-trn and RMSE-test) stands for training and testing data, respectively.

Tables 5–8 show the comparative performance of RKPFCM and RKPFCM-PSO with different existing algorithms such as FCM, G-K, Fuzzy Model Identification (FMI), FCRM, and MFCRM-NC. It is clearly seen from the results that our algorithm RKPFCM-PSO gives the best performance in noisy environments. The local linear models identified by RKPFCM-PSO are given as follows:

\[
y_1(k) = 0.8562y(k - 1) + 0.0548y(k - 2) + 2.0967u(k) - 1.6609u(k - 1) + 0.0814,
\]

\[
y_2(k) = 0.8058y(k - 1) - 0.0924y(k - 2) + 1.3488u(k) - 0.3718u(k - 1) + 0.8032,
\]
5.2.3. Example 3. We consider the Box Jenkins gas furnace data set which is used as a standard test for identification techniques. The data set is composed of 296 pairs of input-output measurements. The input “\(u\)” is the gas flow rate into a furnace and the output “\(y\)” is the CO\(_2\) concentration in the outlet gases. In order to take all the above-mentioned issues into account, we simulated the following experimental case [16]:

all the 296 data pairs are used as training data and \(\{y(k-1), u(k-4)\}\) are selected as input variables to various algorithms, while we use two rules \((C = 2)\). In this example the used parameters are \(a = 1, b = 2, m = 2, G = 10, NP = 30, D = 3, w_{\text{max}} = 0.9, w_{\text{min}} = 0.4, c_1 = 1.95, c_2 = 2.1, \sigma = 10, \text{iter}_{\text{max}} = 1000, \text{and } \varepsilon = 10^{-9}\).

The simulation result given by the RKPFCM-PSO algorithm is illustrated in Figure 6.

Based on the comparison presented in Table 9, it is clear that the proposed algorithm RKPFCM-PSO is more robust to noise than the other algorithms found in literature.

When we use our algorithm RKPFCM-PSO the local linear models identified are given as follows:

\[
y_1(k) = 0.6777y(k - 1) - 0.9593u(k - 4) + 17.3007, \\
y_2(k) = 0.5560y(k - 1) - 1.3297u(k - 4) + 23.5880.
\] (52)

6. Conclusion

In literature, various clustering algorithms have been proposed for nonlinear systems identification. In this work, we developed a new clustering algorithm called RKPFCM-PSO for the nonlinear systems identification. Our algorithm is an improvement of the Possibilistic Fuzzy C-Means Clustering (PFCM) where we used a hyper tangent kernel function to calculate the distance of data point from the cluster centers and a heuristic search algorithm PSO to reach the global minimum of the objective function. The proposed algorithm provides better results of fuzzy modeling of unknown nonlinear systems. The robustness and the quality of this proposed method are demonstrated by simulation results of noisy nonlinear systems described by recurrent equations and application to a Box Jenkins gas furnace system. Thus, the proposed methods show favorable results in noisy environments compared with the techniques mentioned in the literature.

In the future, we will integrate other optimization methods such as the gravitational search algorithm to optimize our hybrid method and we will apply this algorithm for
identification of some complex nonlinear real systems as the robotic or the mechatronic systems.

Conflicts of Interest
The authors declare that they have no conflicts of interest.

References