Research Article

New Approaches to Identification of PWARX Systems

Zeineb Lassoued and Kamel Abderrahim

Numerical Control of Industrial Processes, National School of Engineers of Gabes, University of Gabes, Omar Ibn-Khattab Street, 6029 Gabes, Tunisia

Correspondence should be addressed to Zeineb Lassoued; zeineb.lassoued1@gmail.com

Received 4 May 2013; Revised 8 August 2013; Accepted 14 August 2013

Academic Editor: Hamid Reza Karimi

Copyright © 2013 Z. Lassoued and K. Abderrahim. 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.

We consider the clustering-based procedures for the identification of discrete-time hybrid systems in the piecewise affine (PWA) form. These methods exploit three main techniques which are clustering, linear identification, and pattern recognition. The clustering method based on the $k$-means algorithm is treated in this paper. It consists in estimating both the parameter vector of each submodel and the coefficients of each partition while knowing the model orders $n_a$ and $n_b$ and the number of submodels $s$. The performance of this approach can be threatened by the presence of outliers and poor initializations. To overcome these problems, we propose new techniques for data classification. The proposed techniques exploit Chiu’s clustering technique and the self-artificial Kohonen neural network approach in order to improve the performance of both the clustering and the final linear regression procedure. Simulation results are presented to illustrate the performance of the proposed method.

1. Introduction

Hybrid systems have received great attention in the last years since the behavior of a broad class of physical systems interacts continuous and discrete-event phenomena. The hybrid system is governed by continuous differential equations and discrete variables. The continuous behavior is the fact of the natural evolution of the physical process, whereas the discrete behavior can be due to the presence of switches, operating phases, transitions, computer program codes, and so forth. Several classes have been proposed in the literature for the representation of hybrid systems such as jump linear models (JL models) [1], Markov jump linear models (MJL models) [2], Mixed Logic Dynamical models (MLD models) [3, 4], Max-Min-Plus-Scaling systems (MMPS models) [5], Linear Complementarity models (LC models) [6], Extended Linear Complementarity models (ELC models) [7], and Piecewise Linear models (PWA models) [8, 9]. Only the PWA models are considered in this paper. These models are obtained by decomposing the state-input domain into a finite number of nonoverlapping convex polyhedral regions and by associating a simple linear or affine model to each region. This class of hybrid systems offers several interesting advantages. Firstly, it can approximate any nonlinear system with arbitrary accuracy [10]. Moreover, the properties of equivalence between PWARX models and other classes of hybrid systems allow transferring the results of PWA models to these classes [11]. Therefore, the notion of PWA models can be used to represent complex nonlinear continuous systems. In fact, we can exploit the “divide to reign” strategy which consists in decomposing the domain range of the nonlinear system into a set of operating regions. For each operation region, a linear or affine model is associated. So, the considered complex nonlinear system becomes by modeling as a hybrid system switching between linear subsystems. The analysis and control of PWA systems, like any other type of dynamic system, require a mathematical model of its behavior. This model can be defined through a detailed analysis of the phenomena described by the system using the various laws that govern its operation. This approach can lead to very complicated models that cause problems of exploitation and implementation. However, for engineering, a mathematical model must provide a compromise between accuracy and simplicity of operation. A solution to this problem consists in using the identification approach which allows to build a mathematical model from observed input-output data. In the case of PWARX systems, the identification problem is known to be a challenging problem because it involves both
the estimation of the parameters of the affine submodels and the hyperplanes defining the partition of the state-input regression. Several approaches have been proposed in the literature for the identification of PWARX systems. These methods can be classified in numerous categories of solutions such as algebraic solution [12], clustering-based solution [8], Bayesian solution [13], bounded-error solution [14], and sparse optimization solution [15, 16]. The clustering-based solution has been the most popular because it is characterized by its capacity for modeling complex systems and its simplicity of implementation. It uses the following steps to identify the parameters and the hyperplanes:

(i) constructing small data set from the initial data set,
(ii) estimating a parameter vector for each small data set,
(iii) classifying the parameter vectors in s clusters,
(iv) classifying the initial data set and estimate s submodels with their partitions.

It is easy to deduce that data classification represents the main step toward the objective of PWARX system identification because a successful identification of the parameters depends on the correct data classification. The early approaches use classical clustering algorithms for the data classification [8, 17, 18]. These approaches are characterized by their simplicity of computation and implementation. But they can converge to local minima in the case of poor initializations. Furthermore, their performances degrade when the data are contaminated by the presence of outliers in the data to be classified. Obviously, the use of more powerful clustering algorithms can enhance the performance of these methods. In fact, we suggest to improve the performance of this approach by using other algorithms for data classification such as Chiu's algorithm [19] and the self-adapting artificial kohonen neural Network algorithm [20]. These algorithms allow to reduce the effect of outliers. Moreover, they do not need any initialization. This paper is organized as follows. In Section 2, we present the model and its main assumptions. Section 3 recalls the main steps of the identification of PWARX systems based on clustering technique. Section 4 presents the motivation of using the two proposed methods. In Sections 5 and 6, we describe two algorithms for data clustering allowing to resolve the main problems of the existing methods. The performances of the proposed approach are evaluated and compared through simulation results in Section 7. Section 8 concludes the paper.

2. Model and Assumptions

In the following, we address the problem of identification of PWARX model described by

\[
y(k) = f(\varphi(k)) + e(k),
\]

where

(i) \(y(k) \in R\) is the system output,
(ii) \(e(k)\) is the noise,
(iii) \(k\) is the now time index,
(iv) \(\varphi(k)\) is the vector of regressors which belongs to a bounded polyhedron \(H\) in \(R^d\):

\[
\varphi(k) = [y(k-1), \ldots, y(k-n_y), u(k-1), \ldots, u(k-n_u)]^T,
\]

where \(u(k) \in R^{n_u}\) is the system input, \(n_y\) and \(n_u\) are the system orders, and \(d = n_y + n_u(n_u + 1)\).

(v) \(f\) is a piecewise affine function defined by

\[
f(\varphi) = \begin{cases}
\theta_i^T \varphi & \text{if } \varphi \in H_i \\
\vdots & \\
\theta_s^T \varphi & \text{if } \varphi \in H_s,
\end{cases}
\]

where \(\varphi = [\varphi^T, 1]^T\), \(s\) is the number of submodels, \(H_i\) are polyhedral partitions of the bounded domain \(H\), and \(\theta_i \in R^{dT+1}\) is the parameter vector.

The following assumptions are assumed to be verified.

(A1) The orders \(n_y\) and \(n_u\) and the number of submodels \(s\) are known.

(A2) The noise \(e(k)\) is assumed to be a Gaussian process independent and identically distributed with zero mean and finite variance \(\sigma^2\).

(A3) The regions \(H_i\) are the polyhedral partitions of a bounded domain \(H \subset R^d\) such that

\[
\bigcup_{i=1}^s H_i = H, \quad H_i \cap H_j = \emptyset \quad \forall i \neq j.
\]

Problem Statement. Identify the partitions \(H_i\) and the parameter vectors \(\{\theta_i\}\) of the PWARX model using a data set \(\{(y(k), \varphi(k))\}_{k=1}^N\).

3. Identification of PWARX Models Based on Clustering Approach

This section recalls the main steps of the clustering-based approach for the identification of the PWARX models [8, 17].

3.1. Data Classification. For every pair of data \(\varphi(k), y(k)\) we construct a local set \(\rho_k = \{t_1, \ldots, t_{n_p}\}\) containing in ascending order the index \(k\) of \(\varphi(k), y(k)\) and its \((n_p - 1)\) nearest neighbors satisfying

\[
\forall (\tilde{\varphi}, \tilde{y}) \in \rho_k, \quad \|\varphi(k) - \tilde{\varphi}\|^2 \leq \|\varphi(k) - \phi_s\|^2, \quad \forall (\tilde{\varphi}, \tilde{y}) \notin \rho_k.
\]

Among the obtained local sets \(\rho_k\), some may contain only data from the same model as they are called pure local sets, and others can collect data from multiple submodels that are called mixed sets.
The parameter \( n_\rho \) is chosen randomly as \( n_\rho > d + 1 \). It influences decisively on the performance of the algorithm. The optimal value of \( n_\rho \) is always a compromise between two phenomena: the more this parameter is bigger, the more the parameter estimation is improved and the effect of noise is rejected. However, a large value of \( n_\rho \) increases the number of local mixed sets.

For each local set, we can identify an affine model. To accomplish this task, we adopt the least square method to determine the local parameters \( \theta_k \):

\[
\theta_k = \left( \phi_k^T \phi_k \right)^{-1} \phi_k^T Y_k,
\]

(6)

where \( \phi_k = \left[ \varphi(t_{i,1}) \cdots \varphi(t_{i,n_k}) \right]^T \) and \( Y_k = \left[ y(t_{i,1}) \cdots y(t_{i,n_k}) \right]^T \).

Our objective is to classify the vectors \( \theta_k \) in \( s \) separate classes using a suitable classification technique.

In this paper, three techniques of classification are treated: the \( k \)-means algorithms where the classification is done by minimizing a suitable criterion [8, 21] and Chiu’s clustering technique and the self-artificial Kohonen neural network which are detailed in Sections 5 and 6.

3.2. Parameters Estimation. As the obtained data are now classified, it is possible to determine the \( s \) ARX submodels. We can then estimate the parameter vectors of each submodel \( \theta_i, i = 1, \ldots, s \) using the least square method.

3.3. Regions Estimation. The final step is to determine the regions \( H_i \). The methods of statistical learning such as the support vector machines (SVM) offer an interesting solution to accomplish this task [22, 23]. Support vector machines are a popular machine learning method for classification, regression, and other learning tasks. Originally, the SVM approach was addressed to binary classification, and then it has been extended to multiclass classification. This study is still an ongoing research issue [24, 25].

In our case, it is matter of finding for every \( i \neq j \) the hyperplane that separate points existing in \( H_i \) and in \( H_j \). Given two sets \( H_i \) and \( H_j \), \( i \neq j \), the linear separation problem is to find \( w \in \mathbb{R}^d \) and \( b \in \mathbb{R} \) such that

\[
\begin{align*}
    w^T \varphi_i + b &> 0 & \forall \varphi_i \in H_i, \\
    w^T \varphi_j + b &< 0 & \forall \varphi_j \in H_j.
\end{align*}
\]

(7)

This problem can be easily rewritten as a feasibility problem with linear inequality constraints. The estimated hyperplane separating \( H_i \) from \( H_j \) is denoted with \( M_{ij} \varphi = m_{ij} \), where \( M_{ij} \) and \( m_{ij} \) are matrices of suitable dimensions. Moreover, we assume that the points in \( H_i \) belong to the half-space \( M_{ij} \varphi \leq m_{ij} \).

The regions \( H_i \) are obtained by solving these linear inequalities: it is enough to consider the bounded polyhedron [21]:

\[
\left[ M_{i,1} \cdots M_{i,s} M' \right] \varphi \leq \left[ m_{i,1} \cdots m_{i,s} m' \right],
\]

(8)

where \( M \varphi \leq m \) are the linear inequalities describing \( H \).

4. Motivation of Adopting Chiu’s Clustering Technique and the Self-Artificial Kohonen Neural Network

The classification is an important step to achieve the objective of PWARX model identification because successful identification of both submodels and partitions depends on the performance of the used clustering technique. In fact, this problem presents an area of research in which few results have been devoted in the past because most of the existing methods for the identification of PWARX models are based on classical clustering algorithms such as \( k \)-means methods. However, the classical clustering methods even the modified \( k \)-means algorithms allow only to reduce the influence of outliers and the poor initializations. Consequently, they still suffer from many drawbacks which can be summarized as follows.

(i) They depend on the input signal which must be a persistent excitation to permit to the submodels to have a balanced input [26].

(ii) The parameter \( n_\rho \) must have a small value in order to simplify the computation complexity. However, the best results are generally obtained with a high value of \( n_\rho \).

(iii) The \( k \)-means algorithm does not guarantee the convergence toward an optimal cluster, and therefore it can converge to a local minimum. This is due mainly to the randomly initialization step used by this algorithm.

We are interested in using another technique of classification that can identify and eliminate the misclassified points and can avoid the random initializations. As we have said that we will adopt a similar regression scheme to that of the \( k \)-means procedure, we will focus then on a way that can separate the local parameters \( \theta_k, k = 1, \ldots, N \).

Consider, for example, a dispersion of the local parameters as shown in Figure 1, for example, having the following real parameters:

\[
\begin{align*}
    \theta_1 &= [-1 \ 0], \\
    \theta_2 &= [1 \ 0], \\
    \theta_3 &= [3 \ -2].
\end{align*}
\]

(9)

Based on the results presented in Figure 1, it is well noted that the local parameters are scattered in a way that they most often get around the real parameters. Therefore, the existence of natural groupings of data points because of the PWA properties is clearly observed. We find that determining the centers of these groupings is an interesting solution for our identification problem. For this purpose, we find in [27, 28] a simple and effective algorithm proposed by Chiu for data points clustering. Moreover, we find in the self-adapting artificial Kohonen neural network an interesting and effective way for data classification [20].
5. Chiu’s Classification Method for the Identification of PWARX Systems

Clustering of data forms the basis of many modeling and pattern classification algorithms. The purpose of clustering is to find natural groupings of data in a large data set, thus revealing patterns in the data that can provide a concise representation of the data behavior. Chiu proposed in [27, 28] a simple and effective algorithm for data points clustering.

5.1. Principle. Chiu’s classification method consists in computing a potential value for each point from the data set based on its distances to the actual data points and consider each data point as a potential cluster center. The point having the highest potential value is chosen as the first cluster center. The key idea in this method is that once the first cluster center is chosen, the potential of all other points is reduced according to their distance from the cluster center. All points near the first cluster center will have greatly reduced potential. The next cluster center takes then the highest remaining potential value. This procedure of acquiring new cluster center and reducing the potential of the surrounding points repeats until the potential of all points falls below a threshold or until reaching the number of required clusters.

5.2. PWARX System Identification Based on Chiu’s Classification Method. We now present the use of Chiu’s classification method for the identification of PWARX systems. In fact, consider the local parameters obtained by applying the least square method to the grouping obtained by associating with each \( \varphi \) its \((n_p - 1) \) nearest neighbors as it is described in the \( K \)-means procedure. These \( N \) local parameter vectors \((\theta_i, i = 1, \ldots, N)\) picked out by applying (6) are the objective of our proposed classification technique. Thus, we compute a potential value for each parameter vector \( \theta_i \) using the following expression:

\[
P_i = \sum_{j=1}^{N} e^{-\frac{1}{2} r_i^2 |\theta_i - \theta_j|^2},
\]

where \( r_a \) is a positive constant.

The potential of each parameter vector is a function of its distances to all other parameter vectors. Thus, a parameter vector with many neighboring data points will have the highest potential value. The constant \( r_a \) is the radius defining the neighborhood which can be determined by the following expression:

\[
r_a = \frac{\alpha}{N} \sum_{i=1}^{N} \frac{1}{n_p} \sum_{j=1}^{n_p} \left\| \theta_i - \theta_j \right\|
\]

where \( \alpha \) can be chosen such as \( 0 < \alpha < 1 \).

Since from the set of parameter vectors \((\theta_i, i = 1, \ldots, N)\) there are some parameters obtained from mixed local sets, it is clear that we have interest in eliminating them. Equation (10) can be exploited to eliminate the misclassified parameter vectors. As this equation attributes to the outliers a low potential, we can fix a threshold \( y \) under which the parameter vectors are not accepted and then removed from the data set. This threshold is described by the following equation:

\[
y = \min (P) + \beta (\max (P) - \min (P)),
\]

where \( 0 < \beta < 1 \).

After this treatment the set of parameter vectors is filtered and reduced to \((\theta_i, i = 1, \ldots, N')\) \((N' < N)\). Then, from this new data set we select the parameter vector with the highest potential value as the first cluster center. Let \( \theta_1^* \) be this first center, and let \( P_1^* \) be its potential. The potential is then updated by this formula

\[
P_i \iff P_i - P_1^* e^{-\frac{1}{2} r_i^2 |\theta_i - \theta_1^*|^2}.
\]

The parameter vectors near the first cluster center will have then a reduced potential, and so they are unlikely to be selected as the next center. The parameter \( r_b \) is a positive constant that must be chosen larger than \( r_a \) to avoid obtaining closely spaced cluster centers. The constant \( r_b \) is computed using this formula

\[
r_b = \frac{\alpha}{N} \sum_{i=1}^{N'} \max_{j=1}^{n_p} \left\| \theta_i - \theta_j \right\|
\]

In general after obtaining the \( k \)th cluster center, the potential of every parameter vector is updated by the following formula:

\[
P_i \iff P_i - P_k^* e^{-\frac{1}{2} r_i^2 |\theta_i - \theta_k^*|^2},
\]

where \( P_k^* \) and \( \theta_k^* \) are, respectively, the potential and the center of the \( k \)th parameter vector.

This work is then repeated until obtaining \( s \) potential and \( s \) centers. The \( s \) obtained centers are our sought parameter vectors.
Data: Dispose of $[\tilde{\theta}_i]_i^{N}$ from a given data set $(\varphi_i, y_i)$
Main steps:
(i) Compute $P_i$ for every $[\tilde{\theta}_i]_i^{N}$ according to (10)
(ii) Determine the filtered data points $[\tilde{\theta}_i]_i^{N'}$ ($N' < N$)
(iii) Compute the first cluster center $\tilde{\theta}_1^*$ from (10)
repeat
    Compute the other cluster centers according to the updated potential formula (15)
until obtaining $s$ clusters;
Result: Determination of the parameters $[\tilde{\theta}_i]_i^{N}$

Algorithm 1

Now, after obtaining $s$ centers it matters to search the elements belonging to each cluster. So, we calculate the distance between the estimated output and the real one and classify $\varphi(k)$ into the cluster whose distance is the minimum:

$$\operatorname{argmin} \left( \tilde{\theta}_i^* \varphi_k - y_k \right), \quad i = 1, \ldots, s.$$  \hfill (16)

5.3. Algorithm. See Algorithm 1.

5.4. Properties. The new clustering technique has several interesting properties which can be summarized as follows.

(i) This method does not require any initialization of centers. Therefore, the problem of convergence towards local minima is overcome.

(ii) This method removes the misclassified parameter vectors $(\tilde{\theta}_i)$ from the data set and repeats the overall identification procedure on the reduced set of data points. The outliers can be removed thanks to (10) that associates a low potentials with these parameter vectors.

(iii) The choice of the parameter $n_p$ is more flexible. In fact, we can improve the performance with high value of $n_p$.

6. PWARX Identification Using a Self-Adapting Artificial Kohonen Neural Network

6.1. Principle. The Kohonen neural network is an interesting and effective tool for data classification [20].

The self-organizing Kohonen map is an oriented artificial neural network, consisting of two layers. In the input layer, the neurons correspond to the variables describing the observations. The output layer is generally organized as a grid (map) of neurons with two dimensions. Each neuron represents a group of similar observations.

The Kohonen network is a technique for automatic classification (clustering, unsupervised learning). The objective is to produce a group so that the members in the same cluster are similar and the members located in different clusters are different.

The new clustering technique has several interesting properties which can be summarized as follows.

The new clustering technique has several interesting properties which can be summarized as follows.

The neural network used in the proposed method is formed by one input layer of $p$ neurons and by one output layer of $n$ neurons. The architecture of this network is given by Figure 2 [29]. Each neuron of the Kohonen card receives $p$ signals coming from the input layer. The weight $w_{jm}$ is relative to the connection between the input neuron $j$ and the output neuron $m$. The weight vector $W_i$ associated with neuron $i$ is then composed of $p$ elements.

A Kohonen card computes the euclidian distance between an input $Y$ and its weight vector $W$.

Kohonenn learning uses a function $\eta$, whose value $\eta(i, k)$ represents the strength of the coupling between neuron $i$ and neuron $k$ during the training process. $\eta(i, k) = 1$ for all input neurons $i$ in the neighborhood of neuron $k$ and $\eta(i, k) = 0$ for all other neurons.

6.2. Algorithm. The learning algorithm for Kohonen networks is shown in Algorithm 2.

6.3. PWARX System Identification Based on the Kohonen Neural Network Method. Our purpose is to exploit the Kohonen self-organizing map to identify PWARX systems. Therefore, consider a collection of $N$ data points $(\theta_k, y_k = 1, \ldots, n)$ obtained by (6) of the $k$-means-based procedure. We propose, as a treatment to eliminate the outliers, to apply (10) of the Chiu’s clustering technique. After obtaining the filtered set of data points $\tilde{\theta}_k$, $k = 1, \ldots, N'$ ($N' < N$), the step of data classification will be done by using the Kohonen neural network algorithm while taking an input vector $y_k = \tilde{\theta}_k$.

The output layer is then formed by $s$ neurons ($s$ is the submodels number), and $W_{i}$ are the clusters’ centers.

After obtaining $s$ cluster centers $W_{i}$, we have to define the elements belonging to each original cluster partitioning the regressor input. To perform this task, we calculate the distance between the estimated output and the real one and classify $\varphi_k$ into the corresponding cluster according to the following formula:

$$\operatorname{argmin} \left( W_i^T \varphi_k - y_k \right), \quad i = 1, \ldots, s.$$  \hfill (17)
Data: Dispose of an input vector $Y$
Main steps:
(1) The $n$-dimensional weight vectors $W_1, W_2, \ldots, W_n$ are selected at random.
(2) Each card neuron computes the distance between the weight vector $W_i$ and the input vector $Y$.
(3) The competition between neurons is based on the winner-takes-all strategy. The neuron having the nearest weight vector $W_i$ to the input $Y$ wins the competition. The winner neuron output $\eta(i,k) = 1$ and the other ones are putting then at zero $\eta(i,k) = 0$.
(4) The weight vectors are updated using the neighborhood function and the update rule:
$$W_i \leftarrow W_i + \alpha (Y - W_i) \eta(i,k),$$
where $i = 1, \ldots, n$ and $\alpha$ is a constant verifying $0 < \alpha < 1$.
(5) Stop if the maximum number of iterations has been reached, otherwise continue with step 2.

Algorithm 2

7. Simulations Results

We now present two simulation examples to illustrate the performance of the proposed approaches.

7.1. Quality Measures. Hence, the objective of the simulations is to compare the performance of the proposed methods with that of the modified $k$-means approach. The following quality measures are used to study the performance of each method [30].

(i) The maximum of relative error of parameter vectors is defined by
$$\Delta_{\theta} = \max_{i=1,\ldots,s} \frac{\|\theta_i - \bar{\theta}_i\|_2}{\|\bar{\theta}_i\|_2}, \quad (18)$$
where $\bar{\theta}_i$ and $\theta_i$ are, respectively, the true and the estimated parameter vectors for submodel $i$. The identified model is deemed acceptable if $\Delta_{\theta}$ is small or close to zero.

(ii) The averaged sum of the squared residual errors is defined by
$$\sigma^2_e = \frac{1}{s} \sum_{i=1}^{s} \frac{\text{SSR}_i}{|D_i|}, \quad (19)$$
where SSR$_i = \sum_{y(k) \in D_i} (y(k) - [\varphi(k)] \hat{\theta}_i)^2$ and $|D_i|$ is the cardinality of the cluster $D_i$.

The identified model is considered acceptable if $\sigma^2_e$ is small and/or close to the expected noise variance of the true system.

(iii) The percentage of the output variation, that is explained by the model, is defined by
$$\text{FIT} = 100 \cdot \left(1 - \frac{\|\hat{y} - y\|_2}{\|y - \bar{y}\|_2}\right), \quad (20)$$
where $\hat{y}$ and $y$ are, respectively, the estimated and the true outputs’ vectors and $\bar{y}$ is the mean value of $y$.

The identified model is considered acceptable if FIT is close to 100.

7.2. Example 1. Consider the following PWARX system [17]:
$$y(k) = \begin{cases} 
[-1 & 0] [u(k-1) 1]^T + e(k) & \text{if } \varphi \in [-4 0] \\
[1 & 0] [u(k-1) 1]^T + e(k) & \text{if } \varphi \in [0 2] \\
[3 & -2] [u(k-1) 1]^T + e(k) & \text{if } \varphi \in [2 4],
\end{cases}$$
(21)
where $s = 3$, $n_a = 0$, $n_b = 1$, $H = [-4 4]$, the input $u(k) \in R$ is generated randomly according to the uniform distribution on $H$, $e(k)$ is a white Gaussian noise of variance $\sigma^2 = 0.05$, and $\varphi(k) = [u(k-1)]$ is the regressor vector.
We evaluate the performances of the proposed algorithms (Chiu-based algorithm and Kohonen-based algorithm) and the $k$-means algorithm by using the same identification data.

Figure 3 presents the input and the real output of the system.

The parameter $n_\rho$ defining the cardinality of the local data sets is chosen as follows: $k$-means algorithm ($n_\rho = 6$), Chiu's-based algorithm ($n_\rho = 15$), and Kohonen-based algorithm ($n_\rho = 20$). Taking into account the parameter $n_\rho$ appropriately chosen, each algorithm generates a sequence of local parameters. These local parameters are then classified into three sets, and the center of each set is also determined as it is shown in Figure 4. The centers of each set are depicted by the star symbols.

Based on the results presented in Figure 4, we observe that the outliers are removed by the proposed methods. However, the modified $k$-means method has preserved the outliers.

After obtaining the estimated parameter vectors, we apply the SVM algorithm in order to estimate the regions. We can then attribute each parameter vector to the corresponding region where it is defined.

The estimated output obtained with three algorithms is presented in Figure 5, and the estimated parameter vectors are illustrated in Table 1.

Table 2 presents the quality measures of (18), (19), and (20) of the two proposed methods and the $k$-means approach.

Based on the results presented in Tables 2, 1, and Figure 5, we observe that the proposed methods give better performances than the $k$-means method. The reason is that the proposed method reduces the influence of outliers and does not require any arbitrary initialization.

7.3. Example 2. Consider the following PWARX model [31]:
$$y(k) = \begin{cases} 
0.4 & 0.5 & 0.3 \varphi(k) + e(k) & \text{if } \varphi(k) \in H_1, \\
-0.7 & 0.6 & -0.5 \varphi(k) + e(k) & \text{if } \varphi(k) \in H_2, \\
0.4 & -0.2 & -0.2 \varphi(k) + e(k) & \text{if } \varphi(k) \in H_3,
\end{cases}$$
(22)
where $s = 3$, $n_a = 1$, $n_b = 1$, and $\varphi(k) = [y(k-1) u(k-1)]^T$ is the regressor vector.
Figure 3: Input and real outputs.

Figure 4: Local parameters separated into three sets.

Figure 5: Estimated output with three algorithms.
Consider

\[ H_1 = \{ \varphi \in \mathbb{R}^2 : [1 \ 0.3 \ 0] \varphi \geq 0, [0 \ 0.5 \ 0] \varphi < 0 \}, \]
\[ H_2 = \{ \varphi \in \mathbb{R}^2 : [1 \ 0.3 \ 0] \varphi \leq 0, [1 \ -0.3 \ 0] \varphi < 0 \}, \]
\[ H_3 = \{ \varphi \in \mathbb{R}^2 : [1 \ -0.3 \ 0] \varphi \geq 0, [0 \ 0.5 \ 0] \varphi \leq 0 \}. \]

\hspace{1cm} (23)

The input signal \( u(k) \) and the noise signal \( e(k) \) are random sequences from the normal distribution with variances, respectively, 0.5 and 0.05.

For the k-means algorithm, the parameters of the affine submodels are estimated by minimizing the criterion function. Therefore, the optimization algorithm has the drawback of getting trapped in a local minimum, and poor results can be obtained. In addition, all submodels must have a balanced excitation, but this is not always guaranteed. Thus, we cannot apply the Monte Carlo simulation to the k-means algorithm. Only the algorithm based on Chiu’s clustering technique and the Kohonen neural network-based one are considered in this example.

We carry on this model a Monte Carlo simulation of size 100 with different noise realizations and different input excitations. The size of data generated in each simulation is \( N = 250 \).

We follow the same procedures described above. The estimated parameter vectors are illustrated in Table 3.

The quality measures are computed by the mean of the 100 measures of each simulation. They are presented in Table 4.

### 8. Conclusion

In this paper, we have considered only the clustering-based procedures for the identification of PWARX systems. We focused on the most challenging step which is the task of classification of data points.

The clustering-based procedures require that the model orders \( n_p \) and \( n_b \), and the number of submodels \( s \) is a priori fixed. The parameter \( n_p \), defining the cardinality of the local data sets, is the main tuning knob.

The clustering method based on the \( k \)-means algorithm treated in this paper showed that it performs poorly if the number of mixed local data sets is high. The increase of the number of mixed local data sets which depends on the chosen parameter \( n_p \) leads to the presence of outliers. Added to that the poor initializations lead the algorithm to converge to local minima.

To overcome these problems, we have proposed two techniques of classification. The first is Chiu’s clustering technique and the second is the Kohonen neural network-based one. The proposed methods can guarantee optimal classification.

The main problem to deal with is the classification data points that are consistent with more than one submodel, namely, data points lying in the proximity of the intersection of two or more submodels. Wrong attribution of these data points may lead to misclassifications when estimating the polyhedral regions.

Finally, the choice of persistently exciting input signals for identification (i.e., allowing for the correct identification of all the affine dynamics) is another important topic to be addressed. Moreover, when dealing with discontinuous PWARX models, the choice of the input signal should be such that not only all the affine dynamics are sufficiently excited but also accurate shaping of the boundaries of the regions is possible.
Acknowledgment

This work was supported by the Ministry of the Higher Education and Scientific Research in Tunisia.

References

[1] X. Feng, K. A. Loparo, Y. Ji, and H. J. Chizeck, “Stochastic stability properties of jump linear systems,” IEEE Transactions on Automatic Control, vol. 37, no. 1, pp. 38–53, 1992.

[2] A. Doucet, N. J. Gordon, and V. Krishnamurthy, “Particle filters for state estimation of jump Markov linear systems,” IEEE Transactions on Signal Processing, vol. 49, no. 3, pp. 613–624, 2001.

[3] A. Bemporad and M. Morari, “Control of systems integrating logic, dynamics, and constraints,” Automatica, vol. 35, no. 3, pp. 407–427, 1999.

[4] A. Bemporad, G. Ferrari-Trecate, and M. Morari, “Observability and controllability of piecewise affine and hybrid systems,” IEEE Transactions on Automatic Control, vol. 45, no. 10, pp. 1864–1876, 2000.

[5] B. De Schutter and T. Van den Boom, “On model predictive control for max-min-plus-scaling discrete event systems,” Tech. Rep. 00-04, Control Systems Engineering, Faculty of Information Technology and Systems, Delft University of Technology, The Netherlands, 2000.

[6] A. van der Schaft and J. M. Schumacher, "Complementarity modeling of hybrid systems," IEEE Transactions on Automatic Control, vol. 43, no. 4, pp. 483–490, 1998.

[7] B. De Schutter, “Optimal control of a class of linear hybrid systems with saturation,” in Proceedings of the 38th IEEE Conference on Decision and Control (CDC ’99), pp. 3978–3983, December 1999.

[8] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, “A clustering technique for the identification of piecewise affine systems,” Automatica, vol. 39, no. 2, pp. 205–217, 2003.

[9] L. Bako, “Contribution à l’identification de systèmes dynamiques hybrides,” 2008.

[10] J.-N. Lin and R. Unbehauen, “Canonical piecewise-linear approximations,” IEEE Transactions on Circuits and Systems I, vol. 39, no. 8, pp. 697–699, 1992.

[11] W. P. M. H. Heemels, B. De Schutter, and A. Bemporad, “On the equivalence of classes of hybrid dynamical models,” in Proceedings of the 40th IEEE Conference on Decision and Control (CDC ’01), vol. 1, pp. 364–369, December 2001.

[12] Y. Tian, T. Floquet, L. Belkoura, and W. Perruquetti, “Algebraic switching time identification for a class of linear hybrid systems,” Nonlinear Analysis: Hybrid Systems, vol. 5, no. 2, pp. 233–241, 2011.

[13] A. L. Juloski, S. Weiland, and W. P. M. H. Heemels, “A Bayesian approach to identification of hybrid systems,” IEEE Transactions on Automatic Control, vol. 50, no. 10, pp. 1520–1533, 2005.

[14] A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino, “A bounded-error approach to piecewise affine system identification,” IEEE Transactions on Automatic Control, vol. 50, no. 10, pp. 1567–1580, 2005.

[15] L. Bako, “Identification of switched linear systems via sparse optimization,” Automatica, vol. 47, no. 4, pp. 668–677, 2011.

[16] L. Bako and S. Lecoeuche, “A sparse optimization approach to state observer design for switched linear systems,” Systems & Control Letters, vol. 62, no. 2, pp. 143–151, 2013.

[17] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, “Identification of piecewise affine and hybrid systems,” in Proceedings of the American Control Conference, vol. 5, pp. 3521–3526, June 2001.

[18] H. Nakada, K. Takaba, and T. Katayama, “Identification of piecewise affine systems based on statistical clustering technique,” Automatica, vol. 41, no. 5, pp. 905–913, 2005.

[19] S. Chiu, “Fuzzy model identification based on cluster estimation,” Journal of Intelligent and Fuzzy Systems, vol. 2, no. 3, pp. 267–278, 1994.

[20] J. Kwiatkowski, M. Pawlik, U. Markowska-Kaczmar, and D. Konieczny, “Performance evaluation of different Kohonen network parallelization techniques,” in Proceedings of the International Symposium on Parallel Computing in Electrical Engineering (PARELEC’06), pp. 331–336, September 2006.

[21] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, “A clustering technique for the identification of piecewise affine systems,” Automatica, vol. 39, no. 2, pp. 205–217, 2003.

[22] C. Williams, Support Vector Machines, School of Informatics, University of Edinburgh, 2008.

[23] R. O. Duda, P. E. Hart, and D. G. Stork, Pattern Classification and Scene Analysis, John Wiley & Sons, 1995.

[24] C.-W. Hsu and C.-J. Lin, “A comparison of methods for multiclass support vector machines,” IEEE Transactions on Neural Networks, vol. 13, no. 2, pp. 415–425, 2002.

[25] J. Weston and C. Watkins, “Support vector machines for multi-class pattern recognition,” in Proceedings of the 7th European Symposium on Artificial Neural Networks, pp. 219–224, 1999.

[26] M. Petreczky and L. Bako, “On the notion of persistence of excitation for linear switched systems,” in Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC ’11), pp. 1840–1847, 2011.

[27] S. Chiu, “An efficient method for extracting fuzzy classification rules from high dimensional data,” Journal of Advanced Computational Intelligence, vol. 1, no. 1, pp. 31–36, 1997.

[28] S. Chiu, “Extracting fuzzy rules from data for function approximation and pattern classification,” in Fuzzy Information Engineering: A Guided Tour of Applications, D. Dubois, Ed., chapter 9, John Wiley & Sons, 1997.

[29] S. Talmoudi, K. Abderrahim, R. B. Abdennour, and M. Ksouri, “Multimodel approach using neural networks for complex systems modeling and identification,” Nonlinear Dynamics and Systems Theory, vol. 8, no. 3, pp. 319–316, 2008.

[30] A. Lj. Juloski, S. Paoletti, and J. Roll, “Recent techniques for the identification of piecewise affine and hybrid systems,” in Current Trends in Nonlinear Systems and Control, pp. 79–99, Springer, 2006.

[31] K. Boukharouba, Modélisation et classification de comportements dynamiques des systèmes hybrides [Ph.D. thesis], Université de Lille, Lille, France, 2011.
Submit your manuscripts at
http://www.hindawi.com