A k-plane Clustering Algorithm for Identification of Hybrid Systems

M. Tabatabaei-Pour¹, K. Salahshoor¹, B. Moshiri ²
¹Department of Automation and Instrumentation
Petroleum university of Technology
Sattarkhan Avenue, Tehran
Iran
²CIPCE, School of Electrical and Computer Engineering
Universtiy of Tehran
North Kargar avenue, Tehran
Iran

Abstract: - A new algorithm for identification of discrete time Hybrid Systems in the Piece-Wise Affine (PWA) form is introduced. This problem involves the estimation of both the parameters of the affine submodels and the partition of the PWA map from data. At the first stage we propose a modified version of the k-plane clustering algorithm proposed in [1] to provide initial data classification and parameter estimation. Then we apply the refinement algorithm proposed in [11] repeatedly to the estimated clusters in order to improve both the data classification and the parameter estimation. The k-plane approach clusters the data in the data space instead of feature space and is computationally very efficient. Also the possible modifications on the algorithm which yield to a recursive version for online identification of PWA Hybrid systems are discussed.

Key-Words: - Nonlinear identification, hybrid systems, clustering, classification

1 Introduction

In recent years, the interest in hybrid systems has grown widely. Hybrid models describe processes that evolve according to dynamic equations and logic rules [2]. There are many frameworks for modelling and control of hybrid systems. PieceWise Affine (PWA) systems represent an attractive model structure especially for identification of hybrid systems due to their universal approximation properties and their equivalencies to several classes of hybrid systems [3, 4]. The problem of identification of hybrid systems in the PWA form using experimental data has investigated a few in recent years. The main intricacy of this problem is that the problem of estimating the regions cannot be decoupled from the identification of each submodel. In [5], an overview of different approaches for the identification of PWA systems is presented. Reference [6], proposes an approach that exploits combined use of clustering in the feature space, linear identification, classification techniques for identification in the PieceWise affine AutoRegressive eXogeneous (PWARX) form. Also the algebraic procedure [7], the adapted weights procedure [8], the Bayesian procedure [9] and the bounded-error procedure [10, 11] have been proposed for identification in this form. In [12], the attention is focused on two subclasses of PWA models, namely Hinging Hyperplanes (HHARX) and Wiener PieceWise affine ARX (W-PWARX). For these classes of models, the identification problem is formulated as a suitable mixed-integer linear or quadratic programming problem, which can be solved for the global optimum.

In this paper, we use the k-plane clustering algorithm proposed in [1], which is proposed for clustering of m given points in an n dimensional real space into k clusters by generating k planes that form a local solution for nonconvex problem of minimizing the sum of squares of the 2-norm distances between each point and a nearest plane. We modify the algorithm in order to reduce the effect of outliers and poor initialization influence. This step provides initial classification and parameter estimation. Then, we use the refinement procedure in [11] in order to improve both the classification of data points and the parameter estimation. Finally, in order to find the shape of the
regions we use, as in [6, 11, 12], two class [13, 14] or multi-class [15, 16] separation techniques.

The paper is organized as follows. The Identification problem is formulated in section 2. In section 3, various steps of the proposed algorithm and two illustrative examples are presented. In section 4, we discuss the proposed procedure highlighting future research and possible modifications in order to estimate also the number of submodels from the data sets.

2 Problem Formulation

A PWA map is defined by the equation

\[
f(x) = \begin{cases} 
  \theta'_1 \begin{bmatrix} x \end{bmatrix} & \text{if } x \in \mathcal{X}_1, \\
  \vdots & \\
  \theta'_s \begin{bmatrix} x \end{bmatrix} & \text{if } x \in \mathcal{X}_s, 
\end{cases}
\]

(1)

where \( \mathcal{X} \subset \mathbb{R}^n \) is a bounded polyhedron, \( \{ \mathcal{X}_i \}_{i=1}^{s} \) is a polyhedral partition of \( \mathcal{X} \) (i.e., \( \bigcup_{i=1}^{s} \mathcal{X}_i = \mathcal{X} \)), \( \mathcal{X}_i \cap \mathcal{X}_j = \emptyset \) if \( i \neq j \) and each region \( \mathcal{X}_i \) and is a convex polyhedron, represented in the form \( \mathcal{X}_i = \{ x \in \mathbb{R}^n : H_i x \leq \theta_i \} \), where \( H_i \in \mathbb{R}^{\ell_i \times n} \), and \( \theta_i \in \mathbb{R}^{\ell_i} \) are parameter vectors. A PWA regression problem amounts to reconstructing a PWA map \( f \) from a finite set of data points \( (x(k), y(k)) \), \( k=1,...,N \) generated by the model

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

(2)

where \( e(k) \) are noise samples which are Gaussian independent identically distributed random variables with zero mean and variance \( \sigma^2 \).

When considering the PWARX description of (2), \( x(k) \), the vector of regressors, is denoted by

\[
x(k) = \begin{bmatrix} y(k-1) & y(k-2) & ... & y(k-n_y) 
\end{bmatrix}^T, \]

\[
u(k-1) = \begin{bmatrix} u(k-1) & u(k-2) & ... & u(k-n_u) 
\end{bmatrix}^T, \]

(3)

where \( y \in \mathbb{R} \) and \( u \in \mathbb{R}^n \) represent the input and the output of the system, respectively, and \( n_y, n_u \) are the model orders. It is obvious that \( \mathcal{X} \in \mathbb{R}^n \), \( n = n_y + m_n_u \).

Assumption 1. The data points are collected from the PWARX model (2) specified by the number of submodels \( \mathcal{S} \), the orders \( n_1, n_2, \) the parameter vectors \( \theta'_i \) and the sets \( \mathcal{X}_i, \mathcal{X}_j, i=1,2,...,\mathcal{S} \).

The difficulty of the identification problem depends on which quantities in the assumption 1 assumed to be known. For example, if we know the set \( \mathcal{X}_i, i=1,2,...,\mathcal{S} \) the problem complexity reduces to that of \( \mathcal{S} \) linear identification problems [6]. But since the regions are unknown the identification problem is a challenging problem because it involves the estimation from data points of both the parameter vectors \( \{ \theta'_i \}_{i=1}^{\mathcal{S}} \) and the regions of the regressor space \( \{ \mathcal{X}_i \}_{i=1}^{\mathcal{S}} \).

Assumption 2. The number of submodels, \( s \), is fixed a priori.

During this paper we use this assumption and in section 4 we discuss how we can estimate \( s \) from the data sets.

3 Identification Algorithm

The proposed procedure consists of three steps.

A. Initialization. Data classification and parameter estimation are performed together by using and modifying the k-plane clustering algorithm.

B. Refinement. Misclassifications are reduced and the parameter estimations are improved using a section of the refinement algorithm in [11].

C. Region estimation. Clusters of regressor vectors are linearly separated via two-class or multi-class linear separation.

Figure 1 depicts the flow diagram of the proposed identification procedure. Throughout the next section the following toy example [6] will be used in order to illustrate the initialization procedure.

Example 1. The data are generated by the following PWARX model

\[
y(k) = \begin{cases} 
  -1 [u(k-1)]' + e(k) & \text{if } u(k-1) = x(k) \in \mathcal{X}_1 = [-4, 0], \\
  1 [u(k-1)]' + e(k) & \text{if } u(k-1) = x(k) \in \mathcal{X}_2 = (0, 2), \\
  3 -2 [u(k-1)]' + e(k) & \text{if } u(k-1) = x(k) \in \mathcal{X}_3 = [2, 4], 
\end{cases}
\]

(4)

Figure 2 depicts the system and 50 samples of the data points with \( \sigma^2 = 0.05 \).

3.1 Initialization

Data classification and parameter estimation are carried out simultaneously by using the modified k-plane clustering algorithm as follows:

Algorithm 1.

1. Start with random \( \theta'_0, \theta'_1, ..., \theta'_\mathcal{S} \), given \( \delta \).
Let $j = 0$, $p = 0$, $\theta^0_{\text{best}} = \theta^0$. REPEAT

2.1. Cluster assignment: assign each point to closest plane $P_k = \{x \mid x \in \mathbb{R}^n, y = \theta^T_k \begin{bmatrix} x \\ 1 \end{bmatrix} \}$ for each $\phi_k = \begin{bmatrix} x(k) \\ 1 \end{bmatrix}$, $k = 1, 2, ..., N$, determine $l(k)$ such that:

$$|y_i - \phi_{l(i)} \theta^0_l| = \min_{l:l=1,2,...,N} |y_i - \phi_l \theta^0_l|$$

2.2. Cluster update: Find a plane $P_l$ that minimizes the sum of the squares of distances to each point in cluster $D_l$: for each $l = 1, ..., s$, let $\phi(l)$ be matrix $N(l) \times n$ with rows corresponding to all $\phi_k$ assigned to cluster $D_l$. Find $\theta^{i+1}_l, \theta^{i+1}_2, ..., \theta^{i+1}_s$ by applying least square method to each cluster $D_l$:

$$\theta^{i+1}_l = [\phi(l) \phi(l)^T]^{-1} \phi(l)^T y(l) \quad \text{for } l = 1, ..., s.$$  

2.3. $j = j + 1$. UNTIL there is a repeated overall assignment of points to cluster planes or a non-decrease in the overall objective function.

3. Data assignment:

3.1 For each data point $(y, x), k = 1, ..., N$:

- IF $|y_k - \phi_l \theta^0_l| \leq \delta$ for only one $l = 1, ..., s$,
  THEN mark it as feasible and assign it to cluster $D_l$.
- IF $|y_k - \phi_l \theta^0_l| \leq \delta$ for more than one $l = 1, ..., s$,
  THEN mark it as undecidable.
- Otherwise mark it as infeasible.

3.2 $\theta^{i+1}_l = [\phi(l) \phi(l)^T]^{-1} \phi(l)^T y(l) \quad \text{for } l = 1, ..., s.$

3.3 $p = p + 1, j = j + 1, \theta^0_{\text{best}} = \theta^0$. UNTIL $\theta^0_{\text{best}} = \theta^{i+1}_{\text{best}}$.

The modified algorithm differs from the original version for the addition of section 3. The original algorithm finds $s$ cluster planes in $\mathbb{R}^n$ $P_l = \{x \mid x \in \mathbb{R}^n, y = \begin{bmatrix} x \\ 1 \end{bmatrix} \theta_l \}$ that minimize the sum of squares of distances of each data point to a nearest plane $P_l$. The mentioned algorithm finds a cluster assignment that is locally optimal. The local minimum that traps the algorithm, in many times, have a form like Figure 3, such that two submodels are classified approximately well, but the other one collects the data points from two different submodels. In order to cope with this problem, we add section 3 which make a distinction among infeasible, undecidable, feasible data points. Infeasible data points are not consistent with any submodels and may be outliers. Undecidable data points are consistent with more than one submodel. We discard them during the next parameter estimation (section 3.2) and it will help to reduce misclassifications. For example, in Figure 3 the data points in the intersection of rectangles $r_1$ and $r_2$ will be marked as undecidable and the algorithm neglects them when calculating $\theta^{i+1}_l$ in section 3.2.
procedure helps the algorithm to be repeated again by a better initialization. So the algorithm will find a better solution. 

In order to show the improvement of the modification to the original algorithm we use the following example. 

**Example 2.** Let the data be generated by the following PWARX system [10].

\[
\begin{align*}
y_i &= \begin{cases} 
  \langle -0.4 1 1.5 \rangle \phi_i + e_i & \text{if } \langle 4 -1 10 \rangle \phi_i < 0, \\
  \langle 0.5 -1 -0.5 \rangle \phi_i + e_i & \text{if } \begin{bmatrix} -4 & 1 \\ -5 & -1 \\ 6 \end{bmatrix} \phi_i \leq 0, \\
  \langle -0.3 0.5 -1.7 \rangle \phi_i + e_i & \text{if } \begin{bmatrix} 5 & 1 & 6 \end{bmatrix} \phi_i < 0,
\end{cases}
\end{align*}
\]

which \( \phi_i = [y_{i-1} \ u_{i-1}]^{T} \), \( s = 3 \), \( N = 200 \), and the input signal \( u \in [-5, 5] \), with zero-mean Gaussian noise \( \sigma^2 = 0.05.1000 \) independent runs of the two algorithms are carried out. As shown in table 1 the original and the modified algorithms are trapped in local minimums 652 and 242 times respectively. So the modified algorithm reduces the effect of poor initialization very well.

The bounded error procedure [11], instead of k-plane clustering uses a modified greedy algorithm for initial data classification and parameter estimation. The mentioned procedure has too many tuning parameters which finding a right combination of them is difficult, however this method is very useful when there is no a priori knowledge on the physical system or when one needs to identify a model with a prescribed bounded prediction error [18]. The k-plane clustering algorithm is more computationally efficient and there are a few tuning parameters. Table 2 illustrate the average computation time over 1000 runs of the two algorithms. Moreover, if there is sufficient physical insight into the system and, hence, we can choose appropriate initial parameters the algorithm will be more efficient because of good initialization.

Unfortunately all of the mentioned algorithms for the identification of PWA systems [5, 6, 7, 8, 9, 10, 11, 12] are in batch mode which is a significant limitation. But the proposed algorithm can be written in a recursive version which is useful for online identification of hybrid systems. In order to convert the batch mode k-plane to a recursive version we can use the recursive least squares algorithm in section 3.2 of the algorithm 1 instead of the least squares method. So the algorithm will be adapted for online identification.

### Table 1: Results of 1000 runs of the modified and the original version of the k-plane clustering algorithm for example 2.

<table>
<thead>
<tr>
<th>Algorithm Terminates in</th>
<th>Original</th>
<th>Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global or good local minimum</td>
<td>348(times)</td>
<td>758(times)</td>
</tr>
<tr>
<td>Bad local minimum</td>
<td>652(times)</td>
<td>242(times)</td>
</tr>
</tbody>
</table>

### Table 2: A comparison between average computation time of the Modified k-plane Clustering and the Bounded-Error Algorithm

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Comp. time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bounded-Error</td>
<td>2.864</td>
</tr>
<tr>
<td>k-Plane Clustering</td>
<td>0.0998</td>
</tr>
</tbody>
</table>

### 3.2 Dealing with undecidable data

Discarding undesirable data may work well in many cases. But when there are a large number of data points, discarding them yields to losing a lot of useful information. As an alternative, each undecidable data point could be associated to a submodel that minimizes the identification error. This criterion also may lead to misclassification [17]. Here as in [11] we exploit spatial localization of the regression vectors in order to associate undecidable data points to submodels. For detailed description of this algorithm we refer the interested reader to [11].

After applying the refinement algorithm to the example 1’s clusters obtained from algorithm 1 with \( c = 3 \), there will be no misclassification and no undecidable data point. The finally estimated parameter vectors are:

\[
\begin{align*}
\hat{\theta}_1 & = [-1.0118 - 0.0412], \\
\hat{\theta}_2 & = [3.0297 - 1.8720],
\end{align*}
\]
\[ \theta' = [0.9321 - 0.0801]. \]

That provides a good approximation.

Example 2. (Cont'd) Consider the example with a uniformly distributed noise in \([-0.1, 0.1]\). By choosing \(\delta = 0.1\), the main algorithm stops after 15 iterations and there is only 5 undecidable data points. Therefore, even by neglecting them we could obtain a good result and there is no need to refine the clusters, however the refinement algorithm (with \(c = 5\)) will be terminated after 3 iteration. The final estimated parameter vectors are as follows.

\[ \theta' = [-0.4033 0.9999 1.4905] \]
\[ \theta' = [0.4980 -1.0039 -0.4947] \]
\[ \theta' = [-0.2982 0.4984 -1.7059]. \]

The PWA map and the identification results are depicted in Figure 4.

### 3.3 Region Estimation

During previous steps we have obtained an estimate of each affine submodel in the PWARX representation. The last step of the identification procedure is to obtain an approximation for the unknown regions \(\{\mathcal{A}_1\}_{i=1}^s\) such that \(x_i \in \mathcal{A}_r\) if \((y_i, x_i) \in \mathcal{D}_r\). This problem is equivalent to that of separating \(s\) sets of points by means of linear classifiers (hyperplanes), which is widely investigated in the literature. Support Vector Machines (SVM) [13] and Robust Linear Programming (RLP) [14] methods can be used. Also, when the presence of 'holes' in the model can not be accepted, the extension of the above methods to multi-class cases (M-SVM and M-RLP) [15, 16] can be employed. For detailed description of these methods we refer the interested reader to [17].

Example 1(cont'd). By using linear SVM method, the following estimated hyperplanes are obtained.\[
\begin{align*}
h_1 &= [4.0275 -0.9924 10.3010], \\
h_2 &= [5.1200 1.0026 -6.0073],
\end{align*}
\]
that are very similar to the true ones.

In order to modify the proposed algorithm in recursive version, we should use an incremental method for region estimation. For example we can use incremental linear classifier method such as the Rosenbelat's perceptron algorithm [19]. The proposed modifications are currently under investigation.

### 4 Conclusion and Further Research

We proposed an algorithm for identifying of hybrid systems in the PWARX form. The proposed algorithm consists of tree steps. At the first step, we used a modified k-plane clustering algorithm for initial data classification and parameter estimation. Then a refinement algorithm applied to data clusters in order to improve the data classification and the parameter estimation and finally, we estimated the partition using linear classification techniques.

Modifications on the algorithm which yields to an online method are discussed and are currently under investigation.

During this paper we exploited assumption 2. However, if the number of submodels is unknown, it can be estimated using the rank condition on the data derived in [7] which is currently under investigation. The k-plane clustering algorithm is not guaranteed to converge to optimal minimum and may be trapped in the local minimum. Future researches will focus on the modification on this algorithm to be less sensitive to the initialization.

The problem of choosing an appropriate input signal for identification of hybrid system is an open issue for further research. The main difficulty of this problem is that the input signal should be designed such that all reachable modes are excited sufficiently.
References:


