A Split and Merge Algorithm for Identification of Piecewise Affine Systems

R. S. Baptista, J. Y. Ishihara and G. A. Borges

Abstract— This paper adresses the identification of a class of hybrid dynamical systems which can be represented by a Piecewise Affine Autoregressive Exogenous (PWARX) model. These systems are composed of an usually unknown number of ARX sub-models, each of which corresponds to a polyhedral region of the regression space. It is proposed a Split and Merge clustering algorithm, used under a clustering based identification framework, to estimate the correct number of sub-models. The main advantages of this clustering algorithm is that it requires no initialization and there is only one tuning parameter to be adjusted. The resulting identification procedure is applied in a practical example in the identification of a DC motor with dead zone and saturation.

I. Introduction

In recent years, there has been a growing interest in hybrid systems. Hybrid models are used to describe systems that evolve according to the interaction of continuous and discrete dynamics, i.e., differential equation and logical rules [1], [2]

Piecewise Affine (PWA) systems represent an attractive model structure for the identification of hybrid systems due to their universal approximation properties and their equivalences to several classes of hybrid systems [2], [3].

According to [2], there are four main frameworks for the identification of PWA systems: the clustering approach [3], [4], the Bayesian approach [5], the bounded error approach [6] and the algebraic procedure [7].

The clustering based approach, proposed by [3] consists of four steps: local regression; construction of feature vectors; clustering of the feature vectors; and parameter estimation. Another possible clustering based approach is the one used in [4], which consists of the following steps: clustering measured data; estimating of the boundary hyperplane on regression space; parameter estimation of each sub-model. In both frameworks, clustering is the most crucial step, since it determines the number of modes and the points which will be used to estimate their parameters.

Determining the number of clusters in a data set is a fundamental problem in cluster analysis and numerous ad hoc methods have been suggested for it (see e.g. [8] and references therein). These methods can be classified in two main groups: one which assumes that the elements of the dataset follow a known probability distribution and other that assumes a known geometrical dispersion. However, in the context of identification of PWA systems, the generated data

All the authors are with the Automation and Robotics Laboratory (LARA), Department of Electrical Engineering, University of Brasilia, Brazil. Email: `robertobaptista@unb.br`,
`ishihara@ene.unb.br`, `gaborges@unb.br`

form a group of disjoint clouds with sparse outliers whose elements rarely meet either assumption.

This paper focuses on estimating the number of sub-models in the PWARX model, which is one of the key problems for the identification of PWA systems based on the clustering approach. Most authors assume that the number of sub-models is given *a priori*, a situation in which well known clustering techniques can be used [2], [3]. More recently, [4], [9] proposed clustering procedures to automatically determine the number of sub-models in a PWA system. However, these methods either rely on multiple runs in order to obtain the best estimate according to a validation criteria and/or are very sensitive to initialization. To overcome these drawbacks this work proposes a clustering algorihtm, based on the split and merge concept, that automatically identifies the number of clusters in a dataset.

Another aspect of this paper is the use of the technique in a practical example. Though the identification of PWA systems has been applied to real practical examples, such as in [10]–[12], most contributions remains in a theoretical perspective, identifying systems already presented as PWA systems.

This paper is organized as following: in Section II the PWA system identification problem is formulated; in Section III, the identification framework is proposed, in Section IV the proposed technique is applied in a practical example of a DC motor. An analysis of the procedure is carried out in Section V and finally a conclusion is presented in Section VI

II. Problem Formulation

Let $X \subseteq R^n$ be the regression space, and $X_i, i = 1, 2, \ldots, s, \cup_{i=1}^s X_i = X$ and $X_i \cap X_j = \emptyset \, for \, i \neq j$ represent convex polyhedral subsets of X. A PWARX model is defined by

$$y_k = \begin{cases} \theta_1^T \begin{bmatrix} x_k \\ 1 \end{bmatrix} + e_k & if \ x_k \in \mathrm{X}_1, \\ \vdots & \quad k = 1, 2, \ldots, N \quad (1) \\ \theta_s^T \begin{bmatrix} x_k \\ 1 \end{bmatrix} + e_k & if \ x_k \in \mathrm{X}_s, \end{cases}$$

where $\theta_i, i = 1, \ldots, s, \Theta \subseteq R^{n+1}$ is the parameter vector, $y_k \in \boldsymbol{R}^p$, $x_k \in \boldsymbol{R}^n$, and $e_k \in \boldsymbol{R}^p$ are the output vector, the regression vector, and noise at time k, respectively. The regression vector is given by:

$$x_k = \begin{bmatrix} y_{k-1}^T & \cdots & y_{k-n_y}^T & u_{k-1}^T & \cdots & u_{k-n_u}^T \end{bmatrix}^T, \quad (2)$$

where $u_k \in \boldsymbol{R}^m$ is the output at time k, $n = pn_y + mn_u$ with non-negative integers n_y and n_u denoting the delay order of y and u respectively.

For the identification procedure, it is convinient to define the vector containing N output data samples and their regression vector as:

$$z_k = \begin{bmatrix} x_k \\ y_k \end{bmatrix} \in \boldsymbol{R}^{n+p}, \, k = 1, 2, ..., N \quad (3)$$

In this paper it is assumed that the order of each subsystem, n_y and n_u, is known and equal to all submodels and seek to estimate the number of modes s and the parameter matrices $\theta_i, i = 1, 2, ...s$.

Problem Definition: Given a set of points $z_k = (x_k, y_k)$ assuming it was generated by a structure described by the model (1), estimate the correct number of sub-models s, the regions $X_i, i = 1, 2, \ldots, s$ and the parameters matrices $\theta_i, i = 1, 2, ...s$.

III. PROPOSED IDENTIFICATION FRAMEWORK

This paper presents a framework for the identification of PWA systems based on a clustering approach, similar to the one presented by [2], where the major contribution is in phase 2:

Assumption 1: After a mode transition, the system stays in the new mode for at least a minimum time t_w, for which enough samples are available for an ARX identification.

Phase 1- Identify local sub-models: the complete collected input-output dataset is divided in small local datasets. For each local dataset (LD), an ARX model is generated through the standard least squares (LS) method [13], according to the given order n_y and n_u of the system.

Phase 2 Clustering of the sub-model's parameters: similar identified parameters are grouped together through the proposed split and merge algorithm, which also automatically detects the number of modes in the PWA model. The mean of each cluster is taken as each mode's parameter vector.

Phase 3 Partitioning of the Regression Space: with the identified models, the dataset is relabeled to indicate which points in the regression space were generated by each model and the estimation of the boundary hyperplanes which separate each region of the regression space is carried out. This is achieved with the use of Support Vector Machine (SVM) [14].

A. Identification of each sub-model

A LD is built by dividing sequential N elements of the set z_k into sub-sets W_j. Local parameter vectors, $\theta_j^{LS}, j = 1, \ldots, N_{LD}$, are then estimated for each data set W_j by the LS method [13]:

$$\hat{\theta} = (\phi^T \phi)^{-1} \phi^T Y \quad (4)$$

where ϕ is a collection of regressor vectors x_i and Y a collection of the respective y_i vector. It is common to distinguish between LDs containing only elements generated by the same sub-model, which are referred to as pure LDs, otherwise they are called mixed LDs [3]. Mixed LDs are expected to be found when a transition between modes occurs.

B. Clustering Algorithm

Most frameworks for identification of PWA systems based on the clustering approach, rely mainly on classical clustering algorithms and its derivations: the k-means [3], [2], [9], the Expectation Maximization for Gaussian Mixture Models [4] and Single Linkage [15].

Although widely used, these methods are very sensitive to initialization (such as initial points and mean), tuning parameters and the stop condition. Because of this, most application of these techniques require multiple runs from different starting points and with different tuning parameters and comparison between the results of all the runs. This is a tedious process that involves a lot of heuristics and can lead to imprecise results.

This section presets a split and merge algorithm which uses the silhouette index as a validation criteria.

The two most important aspects of the split and merge algorithm proposed are the splitting phase and the validation index. If the splitting is done in a unordered matter, the algorithm may diverge to the point of having clusters with only one point. The validation index is a key feature of the algorithm, given that it is the stopping condition which should yield the correct number of clusters and the best configuration.

1) Splitting : In the first step of the split and merge algorithm, the well-known Kahunen-Loeve transform (KL) is employed, also known as Principal Component Analysis(PCA), described in [16]:

For a given d-dimensional dataset, the mean vector μ and the covariance matrix Σ is computed. The eigenvectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_l$ and the associated eigenvalues $\lambda_1, \ldots, \lambda_l$ are calculated and sorted according to decreasing eigenvalue. Usually there will be just a few large eigenvalues. The eigenvectors associated to these large eigenvalues can be seen as the direction of dispersion of the elements in a dataset, with the largest eigenvalue representing the greatest dispersion and the smallest representing the least dispersion. Taking only the l eigenvectors having the largest eigenvalues and forming a $d \times l$ matrix $\boldsymbol{A}$, it is possible to project the data onto the l-dimensional subspace according to:

$$\boldsymbol{x}' = \boldsymbol{F}_1(\boldsymbol{x}) = \boldsymbol{A}^t(\boldsymbol{x} - \boldsymbol{\mu}) \quad (5)$$

The PCA is used in this algorithm to split a given dataset into two. In this case only the most significant eigenvalue is taken, all data points are projected into one axis with the mean dislocated to the origin, such that the projected points with positive values belong to one group and the ones with negative values to another. Note that for systems of higher

order, it is necessary to consider more than one significant eigenvalue when using PCA.

2) Validation index : In this algorithm the Silhouette [17] is used for cluster validation. As observed in [17], given no a priori information, the silhouette index outperforms other internal validation indices, such as Dunn's indices and Hubert's correlation with distance matrix.

The silhouette index is the average, over all clusters, of the silhouette width of their points. First, given a point x in the cluster C_i and n_i is the number of points in C_i, compute $a(x)$, the average distance between x and all other points in the cluster C_i:

$$a(x) = \frac{1}{n_i - 1} \sum_{y \in C_i, y \neq x} d(x, y) \tag{6}$$

in this work the Mahalanobis Distance is used as the measure $d(x, y)$. Next compute the minimum of the average distance between x and all other points in the other clusters:

$$b(x) = \min_{h=1,...,I, h \neq I} \left[\frac{1}{n_h} \sum_{y \in C_h} d(x, y) \right] \tag{7}$$

The silhouette index of the complete dataset is defined as:

$$S(x) = \frac{b(x) - a(x)}{max[b(x), a(x)]} \tag{8}$$

The silhouette width for each point x ranges from -1 to 1 and represents a degree of fitness associated with the cluster to which it belongs. Values close to -1 indicate that the point is closer, on average, to points of another cluster than the one to which it belongs. On the other hand, values close to 1 indicate that the average distance between x and the other points in the same cluster is smaller, on average, than the distance to points belonging any other cluster. Data-set partitioning resulting in compact and well separated clusters yields silhouette indexes close to 1.

The proposed Split and Merge algorithm is described in Algorithm 1

C. Estimation of the boundary hyperplanes in the regression space

Here a support vector machine (SVM) structure is employed for classifying two adjacent clusters in the regression space and determine the separating hyperplane between them. As suggested by [4], [9], first the adjacency between the labeled sets is tested. The procedure employed here was the Delaunay triangulation [18] which describes the adjacency of the cells in the Voronoi diagram associated with the dataset. In this case two clusters are said to be adjacent if there is at least one branch between two parameter vector in two different clusters. In this work it is assumed that only two clusters were adjacent and there were no holes in the regression space.

Once the adjacent clusters were determined, any SVM could be used to estimate the separating hyperplane, but considering the dataset was generated with noise the sets may not be linearly separable, which will require a so

---

Algorithm 1 Split and Merge

1: The set of clusters C contains only one cluster, $s = 1$, with all the data points, and initially the silhouette index $S(C) = -1$
2: Split cluster C into two new subsets, using the Kahunen-Loeve transform
3: Create a new cluster set C', with the two new subsets
4: Calculate the global silhouette index,$S(C')$, for this new set C'
5: $C = C'$
6: **while** $S(C) < K * S(C')$ **do**
7:    **for** i = 1:s **do**
8:       Split cluster $C(i)$ into two new subsets, using the Kahunen-Loeve transform
9:       Create a new cluster set C', removing $C(i)$ from C and adding the two new subsets to the end of C
10:       Calculate $S(C')$
11:       **if** $S(C) < K * S(C')$ **then**
12:          $C = C', s = s + 1$
13:          Exit FOR
14:       **end if**
15:    **end for**
16: **end while**
17: $S(C) = S(C')$
18: **repeat**
19:    Flag = 0
20:    **for** $i = 1 : s$ **do**
21:       **for** $j = 1 : s, i \neq j$ **do**
22:          Create a new cluster set C', removing cluster $C(i)$ from C and merging with a different cluster $C(j)$ from C
23:          Calculate $S(C')$
24:          **if** $S(C) < S(C')$ **then**
25:             Exit FOR
26:          **end if**
27:       **end for**
28:       **if** $S(C) < S(C')$ **then**
29:          $C = C', s = s - 1$
30:          Flag = 1
31:          Exit FOR
32:       **end if**
33:    **end for**
34: **until** Flag = 0

---

called soft-margin SVM [19]. In this work the Lagrangian Support Vector Machine (LSVM) proposed by [20] is used. This is a fast and simple algorithm based on an implicit Lagrangian formulation of the dual of the standard quadratic programming of a linear support vector machine.

The algorithm used is a direct implementation of the algorithm presented in [20].

IV. NUMERICAL EXAMPLE

The proposed framework was tested with an example inspired by [12], where a DC motor with saturation and dead zone is identified as a PWA system with the help of the Matlab HIT Toolbox. The referred work shows the necessary adjustments for tuning parameters, as well as multiple runs in order to find an adequate PWA model. As it will be shown, this framework requires much less effort and heuristics. This is a very suitable example of a highly nonlinear single input single output (SISO) system with very distinguished behavior depending on the input.

A typical model for a DC motor is given by:

$$\frac{di}{dt} = \frac{V_{app}}{L} - \frac{R}{L}i - \frac{K_\Phi}{L}\omega \tag{9}$$

$$\frac{d\omega}{dt} = \frac{K_\Phi}{J}i - \frac{b}{J}\omega \tag{10}$$

where i is the current in the armature, ω is the angular speed, b is the viscous friction, J is the moment of inertia

for the motor load and K_1 is the armature or emf constant and K_2 the torque constant.

This model was set up in MATLAB's simulink with added dead zone, saturation ($\pm 9V$) which can be seen as a limitation of the actuator, input noise (normal distribution with $\mu = 0, \sigma^2 = 0.0013$), and the following motor parameters: $L = 0.1H$, $J = 0.001 Kg.m^2$, $b = 0.05$, $R = 2\Omega$ and $K_1 = K_2 = K_\Phi = 0.3$, for simplicity.

The identification procedure was similar to the one described in [12]. Two different input output data-sets were collected one for the identification procedure and a second one for validation. For the identification data-set a sine wave with period of $40s$ and amplitude of $12V$ and sample period of $0.001s$ was taken as the input signal, the angular velocity was the measured output of the system. The validation data-set was generated with a sine wave with period of $20s$ and amplitude of $8V$. In both data-sets the input and output signals were normalized as to vary between -1 and 1, as shown in Fig. 1(a) and 1(b) respectively.

The characteristics of this input signal may not seem usual, specially compared to classical identification procedures. However, establishing an adequate input signal for the identification procedure of hybrid systems, including PWA models, remains an open issue [5], [21], [22]. Taking this into account a choice was made to follow a similar procedure proposed in [12].
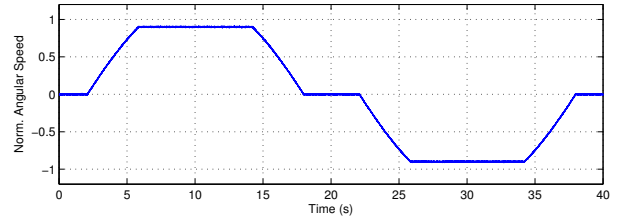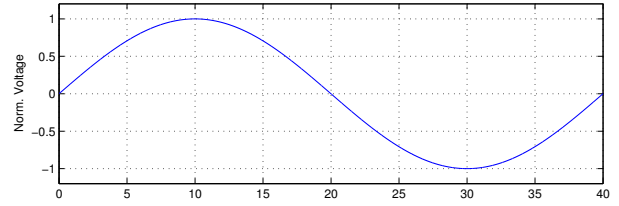
As the first step in the identification procedure the identification dataset was partitioned into sub-sets, each containing 200 elements, $W_j, j = 1, \ldots 200$. Next, for each sub-set an ARX model was obtained generating a new set of parameter vectors θ_j^{LS}. Over this new dataset the split and merge algorithm was applied and the results can be seen in Fig. 2. The algorithm separated the identified parameter vectors in 5 clusters, which as will be latter shown, represents the dead zone, positive dynamical zone, positive saturation zone, negative dynamical zone and negative saturation zone.

For each identified cluster the mean was taken as the parameters representing that sub-model and their values are:
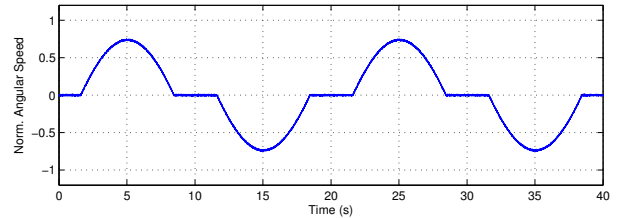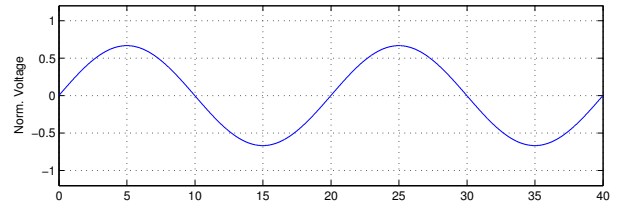
$$
\begin{aligned}
\hat{\theta}_1 &= \begin{bmatrix} 0.0147 & -0.0438 & -0.9377 \end{bmatrix}^T \\
\hat{\theta}_2 &= \begin{bmatrix} -0.0413 & 1.9854 & 0.6463 \end{bmatrix}^T \\
\hat{\theta}_3 &= \begin{bmatrix} -0.0081 & 0.0107 & 0.0048 \end{bmatrix}^T \\
\hat{\theta}_4 &= \begin{bmatrix} -0.0069 & 1.8856 & -0.5990 \end{bmatrix}^T \\
\hat{\theta}_5 &= \begin{bmatrix} 0.0044 & 0.0096 & 0.8869 \end{bmatrix}^T
\end{aligned}
\tag{11}
$$

The next step was to label all the points in the regression space to indicate which sub-model it created. Before applying the LSVM algorithm the sub-sets containing θ_j^{LS} had to be reordered according to the adjacency between the sets with the Delaunay triangulation.

The LSVM algorithm was applied to this labeled dataset in order to detect the hyperplanes which separate each adjacent set in the regression space. The previously estimated parameter vectors are represented by the regions in the subspace according to the following bounding hyperplanes:

(a) I/O Identification Data

(b) I/O Validation Data

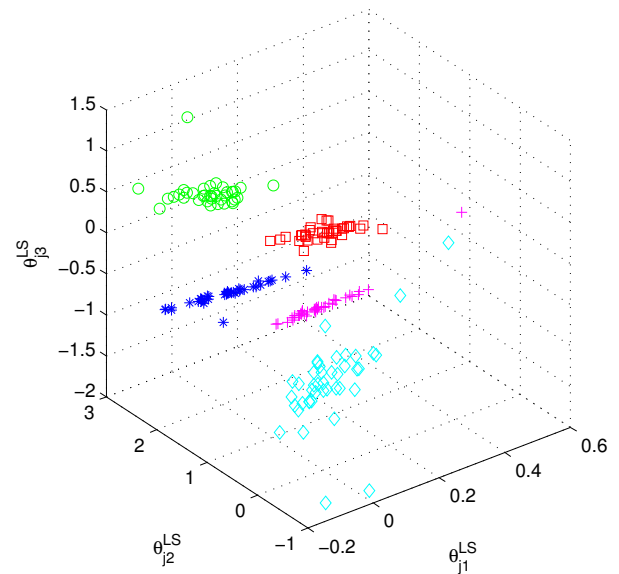Fig. 1. Identification and Validation Data

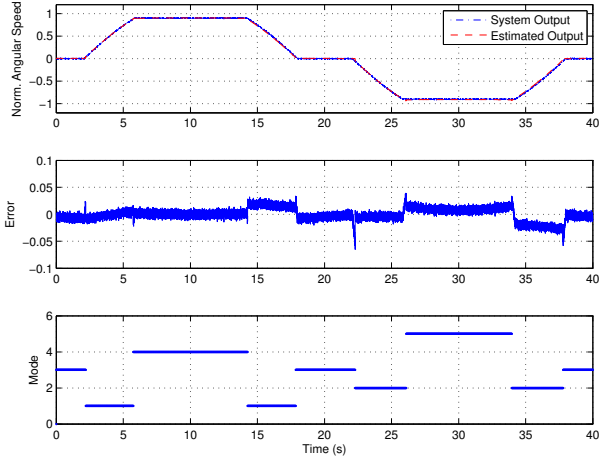Fig. 2. Clustering of Identified Parameters

Fig. 3. Estimated and Measured Output for the Identification Data-set

Fig. 4. Estimated and Measured Output for the Validation Data-set

Data-set	MSE
Identification Data-set	1.103×10^{-4}
Validation Data-set	0.0028

TABLE I

MSE FOR IDENTIFICATION AND VALIDATION DATA-SET

	Initialization	Tuning Parameters
E.M with G.M.M.	(α_i), (μ_i),(Σ_i)	(ϵ)
C-means	(μ_i). (s)	(c), (σ), (ϵ)
Single Linkage	none	(ℓ), (c)
Split and Merge	none	(K)

TABLE II

INITIAL AND TUNING PARAMETERS

$$\hat{h}_{12} = \begin{bmatrix} -5.2962 & 20.2830 & 11.6105 \end{bmatrix}^T$$
$$\hat{h}_{23} = \begin{bmatrix} 3.9342 & 7.5439 & 2.6622 \end{bmatrix}^T$$
$$\hat{h}_{34} = \begin{bmatrix} 3.5837 & 8.2973 & -2.8331 \end{bmatrix}^T$$
$$\hat{h}_{45} = \begin{bmatrix} 0.9595 & 13.6510 & -11.5655 \end{bmatrix}^T$$

Once an identified model is produced, it is necessary to validate it with a different experimental dataset. First to check if the identified model was reasonable, the output of the system subject to the input of the identification dataset was estimated. A close observation to fig 3 indicates that this model is a good approximation for this dataset.

Next the same model was used with the validation dataset. The result can be seen in Fig 4. Table IV shows the mean square error for both cases.

V. ANALYSIS

The results from the previous section shows that the proposed framework is adequate to the identification of PWA models for nonlinear systems. The main advantage of the proposed framework is the clustering algorithm which was able to identify the right number of sub-models in the identification data-set with no *a priori* knowledge.

Most procedures adopted for the automatic estimation of the number of clusters for the identification of PWA models rely on testing validation indices through a different number of clusters and seek for an elbow point, which should indicate a minimum for the tested criteria. There is also the issue of adjusting various parameters as well as setting initial conditions. The necessary adjustments for each algorithm is shown in table (V).
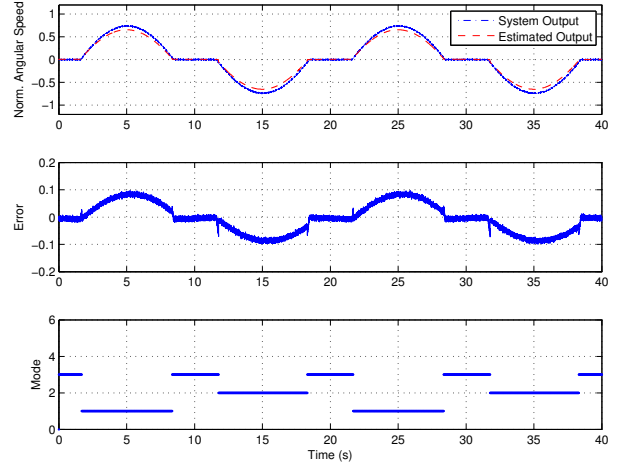
The statistical clustering approach (Expectation Maximization with Gaussian Mixture Models) used in [4] requires the initialization of the weighting parameters (α_i), the initial covariance matrices (Σ_i), the initial mean vectors (μ_i) and the convergence tolerance(ϵ), which implies the need for multiple starts and multiple runs with different given number of clusters. The method based in Fuzzy C-Means clustering and competitive learning proposed by [9], besides the initial mean vectors (μ_i), also requires the tuning of the initial number of clusters(s), the acceptable degree of membership(σ), the convergence tolerance(ϵ) and a constant c which indicates the cardinality of each set.

In the single-linkage procedure used by [15] there is also the need to tune parameters related to the minimal distance between clusters (ℓ) and the cardinality of each set (c). Though this algorithm has only two tuning parameters, they reflect guesses about the resulting clusters.

The work of [12] shows the difficulty in adjusting various parameters in a ready made software kit.

The greatest advantage of this proposed clustering algorithm is the fact that it requires no initialization, and there is only one tuning parameter to be set. Increasing this tuning parameter will stimulate splitting in the first phase of the algorithm. Beyond a threshold if will lead to ill-partitioning of the clusters, resulting in a poor validation (Silhouette) index.

The stop criterion in the SM algorithm is the Silhouette index, which indicates a degree of fitness for the resulting configuration. In this way there is no need to set or guess values for both initialization and validation. The tuning parameter, constant K, is used to seek best results, that is,

the greatest value for the Silhouette index.

Although there is some freedom to adjust the constant K in the SM algorithm, it's value should not be much greater than one. In this example this parameter gave the best result when tunned to 1.15, which yielded the global silhouette index of 0.9787.

Regarding the identification framework, the experimental results shows the correct detection of the number of modes for an adequate PWA model for the identified nonlinear system. Another aspect is the correct grouping of the locally identified parameters vectors θ_j^{LD} since this information was used to estimate the parameter vector for each mode. The size of the LD is another constant which has to be set in this framework, it should be large enough to provide a good estimation of local the local parameter vector, but small enough to generate many points for the clustering algorithm. In this experiment the LDs where formed with 200 points. It was observed that below 100 points the estimation of the various θ_j^{LD} was very poor and above 300 there were a small amount of parameter vectors, negatively affecting the clustering phase.

From the estimated output associated to the validation data-set in figure 4, it is possible to observe a good representation of the nonlinearities features associated with each mode, which indicates a good estimation of the separating hyperplanes. A small gain error is evident comparing the peak values of the estimated and measured output for the validation data-set. This can be a result of the fairly naive approach of taking the mean of each cluster as the parameter vector for each mode. Due to process noise and the identification in transition periods (mixed LD's) some incorrect points are associated with each vector. In this case a robust estimation of the parameters vectors for each cluster could lead to a better model, but overall the proposed framework was able to find a consistent PWA model for a highly nonlinear system.

VI. CONCLUSION

This work presented an algorithm for automatic detection of the number of modes to be used as part of a framework for the identification of PWA systems. Different from previous algorithms of the litterature, the proposed algorithm requires no initialization of parameters and there is only one tuning parameter to be adjusted. This avoids the need for multiple runs in search for the best value of a certain criteria. In fact the algorithm searches for the best clustering configuration of the dataset.

Another contribution of this work is the application to a practical example. The proposed framework was capable of successfully identifying a model of a DC motor with saturation and dead zone, a highly complex nonlinear system.

VII. ACKNOWLEDGMENTS

REFERENCES

[1] M. Tabatabaei-Pour, M. Gholami, K. Salahshoor, and H. R. Shaker, "A Clustering-Based Bounded-Error Approach for Identification of PWA Hybrid Systems," in *IEEE International Conference on Control, Automation, Robotics and Vision*, 2006, pp. 1–6.

[2] S. Paoletti, A. Juloski, G. Ferrari-Trecate, and R. Vidal, "Identification of hybrid systems: A tutorial," *European Journal of Control*, 2007.

[3] 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.

[4] 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, May 2005.

[5] A. Juloski, S. Weiland, and W. Heemels, "A Bayesian approach to identification of hybrid systems," in *IEEE Conference on Decision and Control*, 2004, pp. 13–19.

[6] 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, Oct. 2005.

[7] R. Vidal, S. Soatto, and S. Sastry, "An algebraic geometric approach to the identification of a class of linear hybrid systems," in *IEEE International Conference on Decision and Control*, 2003, pp. 167–172.

[8] Z. He, A. Cichocki, S. Xie, and K. Choi, "Detecting the number of clusters in n-way probabilistic clustering," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, pp. 2006–2021, 2009.

[9] M. Gegundez, J. Aroba, and J. Bravo, "Identification of piecewise affine systems by means of fuzzy clustering and competitive learning," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 8, pp. 1321–1329, Dec. 2008.

[10] A. Juloski, W. Heemels, and G. Ferraritrecate, "Data-based hybrid modelling of the component placement process in pick-and-place machines1," *Control Engineering Practice*, vol. 12, no. 10, pp. 1241–1252, Oct. 2004.

[11] M. Vasak, L. Mladenovic, and N. Peric, "Clustering-based identification of a piecewise affine electronic throttle model," in *Annual Conference of IEEE Industrial Electronics Society, 2005. IECON 2005.* IEEE, 2005, pp. 177–182.

[12] N. Canty and T. O'Mahony, "Design considerations for piecewise affine system identification of nonlinear systems," in *IEEE Mediterranean Conference on Control and Automation*, June 2009, pp. 157–162.

[13] L. Ljung, *System Identification Theory for the User*, 2nd ed. Prentice Hall PTR, 1999.

[14] V. Vapnik, *Statistical Learning Theory*. Wiley-Interscience, 1998.

[15] G. Ferrari-Trecate and M. Muselli, "Single-linkage clustering for optimal classification in piecewise affine regression," in *IFAC Conference on Analysis and Design of Hybrid Systems*, Saint-Malo, 2003.

[16] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley, 2001.

[17] M. Brun, C. Sima, J. Hua, J. Lowey, B. Carroll, E. Suh, and E. Dougherty, "Model-based evaluation of clustering validation measures," *Pattern Recognition*, vol. 40, no. 3, pp. 807–824, Mar. 2007.

[18] H. Edelsbrunner, *Algorithms in Combinatorial Geometry*. Springer, 1987.

[19] V. N. Vapnik, *Statistical Learning Theory*. Wiley, 1998.

[20] O. Mangasarian and D. Musicant, "Lagrangian support vector machine," *The Journal of Machine Learning Research*, vol. 1, no. September, pp. 161—-177, Aug. 2001.

[21] M. Tabatabaei-Pour, K. Salahshoor, and B. Moshiri, "A Modified k-plane Clustering Algorithm for Identification of Hybrid Systems," in *IEEE World Congress on Intelligent Control and Automation*, 2006, pp. 1333–1337.

[22] R. Vidal, "Recursive identification of switched ARX systems," *Automatica*, vol. 44, no. 9, pp. 2274–2287, Sept. 2008.