A Greedy Approach to Identification of Piecewise Affine Models

Alberto Bemporad, Andrea Garulli, Simone Paoletti, and Antonio Vicino

Università di Siena, Dipartimento di Ingegneria dell'Informazione,
Via Roma 56, 53100 Siena, Italy
{bemporad,garulli,paoletti,vicino}@dii.unisi.it

Abstract. This paper addresses the problem of identification of piecewise affine (PWA) models. This problem involves the estimation from data of both the parameters of the affine submodels *and* the partition of the PWA map. The procedure that we propose for PWA identification exploits a greedy strategy for partitioning an infeasible system of linear inequalities into a minimum number of feasible subsystems: this provides an initial clustering of the datapoints. Then a refinement procedure is applied repeatedly to the estimated clusters in order to improve both the data classification and the parameter estimation. The partition of the PWA map is finally estimated by considering pairwise the clusters of regression vectors, and by finding a separating hyperplane for each of such pairs. We show that our procedure does not require to fix a priori the number of affine submodels, which is instead automatically estimated from the data.

1 Introduction

Black-box identification of nonlinear systems has been widely addressed in different contexts. A large number of model classes have been considered and their properties deeply investigated (see the survey papers [1,2] and references therein). In this paper, we deal with the problem of identifying a piecewise affine (PWA) model of a discrete-time nonlinear system from input-output data. PWA systems have become more and more popular in recent years, thanks to their equivalence with several classes of hybrid systems [3,4]. However, estimation of hybrid models from data has not received the attention it deserves in the control community, except for few very recent contributions [5,6,7].

Identification of PWA models involves the simultaneous estimation of both the parameters of the affine submodels *and* the partition of the PWA map. The first issue is closely related to the problem of classifying the data, *i.e.*, the problem of correctly assigning each datapoint to an affine submodel. In [5] a two-phase approach for the classification of the datapoints and the estimation of the parameters has been proposed. The classification problem is reduced to an optimal clustering problem, in which the number of clusters is fixed. Once the datapoints have been classified, linear regression is used to compute the final submodels. In [6] the attention is focused on two subclasses of PWA models, namely hinging

O. Maler and A. Pnueli (Eds.): HSCC 2003, LNCS 2623, pp. 97–112, 2003.
© Springer-Verlag Berlin Heidelberg 2003

hyperplanes (HHARX) and Wiener piecewise affine (W-PWARX) autoregressive exogenous models. For these classes of models, the identification problem is formulated as a suitable mixed-integer linear (or quadratic, depending on the choice of the cost function) programming problem, which can be solved for the global optimum. Also in [7] the identification problem for a class of hybrid systems is formulated as an optimization problem, and an algorithm which provides an approximation of the optimal solution is developed. It makes it possible to incorporate particular a priori knowledge, such as the level of abstraction, the structure, and the desired accuracy of the model.

The identification procedure proposed in this paper does not require that the number of affine submodels is fixed a priori. Hence, this number must be *estimated* from data, together with the parameters of the submodels and the partition of the map. The key approach here is the selection of a bound on the prediction error. This induces a set of linear inequality constraints on the parameters of the PWA model to be estimated. These constraints are generally infeasible (otherwise a single affine model would fit the data within the given error level). Hence, a suitable strategy is suggested for picking a number of submodels which is compatible with the available data and the selected bound. In particular, the greedy strategy proposed in [8] for partitioning an infeasible system of linear inequalities into a minimum number of feasible subsystems, is exploited in order to provide an initial clustering of the datapoints. To each feasible subsystem a set of feasible parameter vectors is then associated according to the bounded-error assumption [9,10]. After the first classification, a projection algorithm is applied repeatedly to the estimated clusters in order to improve both the classification of the datapoints and the estimation of the parameters. In this phase, the datapoints are grouped together according to the fact that they are fitted by the same affine submodel, so that outliers are automatically rejected. Notice that the final number of submodels and the corresponding parameter vectors will depend on the selected bound on the prediction error, so that this determines both the complexity of the model and the quality of the approximation. The partition of the PWA map is finally estimated by considering pairwise the clusters of regression vectors, and finding a separating hyperplane for each of such pairs. Linear Support Vector Machines [11] are suitable for this aim. In this paper, we show that, given two clusters of points, the problem of finding a generalized separating hyperplane (*i.e.*, a hyperplane that minimizes the number of misclassified points) can be formulated as a maximum feasible subsystem problem, for which computationally efficient methods exist [12].

2 Problem Statement

Let $F : \mathcal{X} \mapsto \mathbb{R}^p$ be a nonlinear map defined over the polyhedron $\mathcal{X} \subseteq \mathbb{R}^n$, and assume that a collection of N samples (y_k, x_k), $k = 1, \dots, N$, of $F(\cdot)$ is given, where

$$y_k = F(x_k) + e_k \ , \tag{1}$$

and $e_k \in \mathbb{R}^p$ is a perturbation term. The aim is to find, on the basis of the available samples, a Piecewise Affine (PWA) approximation $f(\cdot)$ of $F(\cdot)$,

$$f(x) = \begin{cases} \theta_1' \begin{bmatrix} x \\ 1 \end{bmatrix} & \text{if } x \in \mathcal{X}_1 \\ \vdots & \vdots \\ \theta_s' \begin{bmatrix} x \\ 1 \end{bmatrix} & \text{if } x \in \mathcal{X}_s \ , \end{cases} \tag{2}$$

where $\theta_i \in \mathbb{R}^{(n+1)\times p}$ are parameter matrices, and $\{\mathcal{X}_i\}_{i=1}^s$ is a polyhedral partition of $\mathcal{X}$ (*i.e.*, $\bigcup_{i=1}^s \mathcal{X}_i = \mathcal{X}$, $\mathcal{X}_i \bigcap \mathcal{X}_j = \emptyset$ if $i \neq j$, and each *region* $\mathcal{X}_i$ is a convex polyhedron, represented in the form[1] $\mathcal{X}_i = \left\{ x \in \mathbb{R}^n : H_i \begin{bmatrix} x \\ 1 \end{bmatrix} \leq 0 \right\}$, where $H_i \in \mathbb{R}^{q_i \times (n+1)}$).

In the context of nonlinear function approximation, y_k represent values of $F(\cdot)$ obtained at certain points x_k, and e_k is either zero (for instance when $F(\cdot)$ can be computed analytically), or an approximation error (for instance when $F(\cdot)$ is evaluated numerically by iterative procedures, as in the case of implicit functions or optimal value functions).

In the context of system identification, $k \in \mathbb{Z}$ is the time index, x_k is the regression vector (accordingly, $\mathcal{X}$ is called the *regressor set*), y_k is the system output, and e_k is noise. For instance, when identifying state-space models of the form

$$\begin{cases} \xi_{k+1} = F_1(\xi_k, u_k) \\ \quad \eta_k = F_2(\xi_k, u_k) \ , \end{cases} \tag{3}$$

x_k contains the components of the state and input vectors at time k, *i.e.*, $x_k = [\xi_k' \ u_k']'$, whereas $y_k = [\xi_{k+1}' \ \eta_k']'$, assuming that the state vector is measurable. A typical reason for estimating a PWA approximation of (3) is for applying the tools of verification, controller synthesis, and stability analysis developed for linear hybrid systems, to nonlinear processes.

In this paper, we rather focus on identification of PWARX (Piecewise affine AutoRegressive eXogenous) models in the form (2), where $p = 1$, the regression vector is defined as $x_k = [y_{k-1} \dots y_{k-n_a} \ u_{k-1}' \dots u_{k-n_b}']'$, and $u_k \in \mathbb{R}^m$ and $y_k \in \mathbb{R}$ denote the system input and output, respectively. In this case, the parameter vectors $\theta_i \in \mathbb{R}^{n+1}$, $i = 1, \dots, s$, contain the coefficients of the ARX *submodels*. For simplicity of exposition, throughout the paper it is assumed $p = 1$, though the presented approach is easily applicable to the case $p > 1$ by small

[1] We do not assume here that $f(\cdot)$ is continuous. Without this assumption, definition (2) is not well posed in general, since the function could be multiply defined over common boundaries of the regions $\mathcal{X}_i$. One could avoid this by replacing some of the "$\leq$" inequalities with "$<$" in the definitions of the polyhedra $\mathcal{X}_i$, although this issue is not of practical interest in the problem at hand.

amendments to the procedures shown in Sections 3 and 4. For a more compact notation, hereafter we will consider the extended regression vector $\varphi_k = [x_k' \; 1]'$.

The key approach of this paper consists in selecting a bound δ on the *prediction error*, *i.e.*, in requiring

$$|y_k - f(x_k)| \leq \delta \,, \quad \forall k = 1, \dots, N, \tag{4}$$

for some $\delta > 0$. Notice that the prediction error is the sum of the approximation error $F(x_k) - f(x_k)$ and the perturbation term e_k. Then, the considered identification problem can be formulated as follows:

Problem 1. Given N datapoints (y_k, x_k), $k = 1, \dots, N$, estimate a positive integer s, a partition $\{\mathcal{X}_i\}_{i=1}^s$ and parameter vectors $\{\theta_i\}_{i=1}^s$, such that the corresponding PWA model (2) of system (1) is compatible with the available data according to condition (4).

Condition (4) naturally leads to a set-membership or bounded-error approach to the identification problem (see, *e.g.*, [9,10]). Notice that the bound δ is not necessarily given a priori, it is rather a tuning knob of the procedure. A reliable choice of it can often be made a posteriori by performing a series of trials for different values of δ, and then selecting a value that provides a good trade-off between the complexity of the model (in terms of number of submodels) and the quality of the approximation (in terms of mean square error). To clarify this, consider the case of nonlinear function approximation, where the smaller δ, the larger the number s of submodels needed to fit the datapoints (y_k, x_k) to a PWA map (2). On the other hand, the larger δ, the worse the approximation, since large errors are allowed.

The following example will be used throughout the paper to illustrate the mechanism of the proposed identification procedure.

Example 1. Let the data be generated by the PWARX system

$$y_k = \begin{cases} \begin{bmatrix} -0.4 \; 1 \; 1.5 \end{bmatrix} \varphi_k + e_k & \text{if } \begin{bmatrix} 4 \; -1 \; 10 \end{bmatrix} \varphi_k < 0 \\ \begin{bmatrix} 0.5 \; -1 \; -0.5 \end{bmatrix} \varphi_k + e_k & \text{if } \begin{bmatrix} -4 & 1 & -10 \\ 5 & 1 & -6 \end{bmatrix} \varphi_k \leq 0 \\ \begin{bmatrix} -0.3 \; 0.5 \; -1.7 \end{bmatrix} \varphi_k + e_k & \text{if } \begin{bmatrix} -5 \; -1 \; 6 \end{bmatrix} \varphi_k < 0 \,, \end{cases}$$

for which $\varphi_k = [y_{k-1} \; u_{k-1} \; 1]'$ and $s = 3$. The input signal u_k and the noise signal e_k are uniformly distributed in $[-5, 5]$ and $[-0.1, 0.1]$, respectively. $N = 200$ estimation datapoints are used. The partition of the regressor set and the set of available regression vectors are depicted in Figure 1. The three regions contain 55, 66 and 79 points, respectively.

3 The MIN PFS Problem

In this section we will describe the greedy algorithm proposed in [8] for partitioning an infeasible system of linear inequalities into a minimum number of feasible

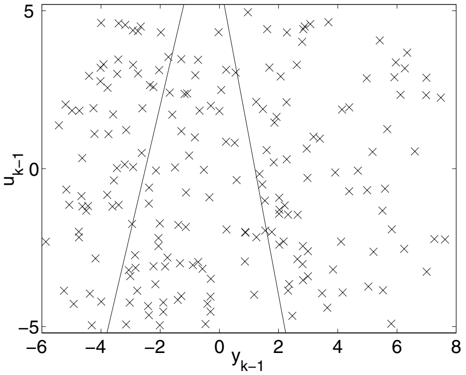

Fig. 1. The partition of the regressor set and the available regression vectors

subsystems. We will also show how to use this algorithm in our identification procedure to obtain an initial classification of the datapoints and a set of feasible parameter vectors for each submodel.

In the first part of our identification procedure, we do not consider the problem of estimating the hyperplanes defining the polyhedral partition of the regressor set. We focus only on classifying the datapoints according to the fact that they are fitted by the same affine submodel. Obviously, in this phase it is reasonable to look for the minimum number of submodels (namely s) fitting all (or most of, due to possible outliers) the datapoints. In other words, we look for the "simplest" PWA model that is consistent with the data and condition (4), where, for a given δ, "simplicity" is measured in terms of the number of affine submodels.

By requiring condition (4), the classification problem can be formulated as follows:

Problem 2. Given $\delta > 0$ and the (possibly infeasible) system of N linear complementary inequalities

$$\begin{cases} \varphi_k'\theta \leq y_k + \delta \\ \varphi_k'\theta \geq y_k - \delta \end{cases}, \quad k = 1, \ldots, N, \tag{5}$$

find a partition of this system into a minimum number s of feasible subsystems, under the constraint that two paired complementary inequalities must be included in the same subsystem (i.e., they must be simultaneously satisfied by the same parameter vector θ).

The above formulation makes it possible to address simultaneously the two fundamental issues of data classification and parameter estimation. Given any solution of Problem 2, the partition of the complementary inequalities provides the classification of the datapoints, whereas each feasible subsystem defines the set of feasible parameter vectors for the corresponding affine submodel.

Problem 2 is an extension of the combinatorial problem of finding a Partition of an infeasible system of linear equalities into a MINimum number of Feasible Subsystems, which is known in the literature as MIN PFS. Since MIN PFS turns out to be NP-hard, and we are only interested in a suboptimal solution of Problem 2 to initialize our identification procedure, we adopt the greedy approach proposed in [8], which efficiently provides good approximate solutions. This approach divides the overall partition problem into a sequence of subproblems. Each subproblem consists in finding a parameter vector $\theta \in \mathbb{R}^{n+1}$ that satisfies as many pairs of complementary inequalities as possible. Starting from system (5), maximum feasible subsystems are iteratively extracted (and the corresponding inequalities removed), until the remaining subsystem is feasible. Due to the suboptimality and randomness of the greedy approach [8], this procedure yields a (not necessarily minimal) partition into feasible subsystems.

The problem of finding one $\theta \in \mathbb{R}^{n+1}$ that satisfies as many pairs of complementary inequalities as possible extends the combinatorial problem of finding a MAXimum Feasible Subsystem of an infeasible system of linear inequalities, which is known in the literature as MAX FS. Based on the consideration that also MAX FS is NP-hard, the approach proposed in [8] tackles the above extension of MAX FS using a randomized and thermal variant of the classical Agmon-Motzkin-Schoenberg relaxation method for solving systems of linear inequalities [13,14]. This provides good solutions in a reasonably short computation time.

3.1 The Randomized Relaxation Method for the MAX FS Problem

We now briefly describe the randomized relaxation method proposed in [8] for solving the extension of MAX FS to the setting with pairs of complementary inequalities[2].

First, the algorithm requires the definition of a maximum number of cycles $C > 0$, an initial temperature parameter $T_0 > 0$, and an initial estimate $\theta^{(1)} \in \mathbb{R}^{n+1}$ (*e.g.*, randomly selected, or computed by least squares). During each cycle all the datapoints are selected in the order defined by a prescribed rule (*e.g.*, cyclicly, or uniformly at random without replacement), so that each cycle consists of N iterations. If k is the index of the selected datapoint, and $\theta^{(j)}$ is the current estimate (where $j = 1, \dots, CN$ is the iteration counter), the corresponding violation is computed as follows:

$$v_j^k = \begin{cases} \varphi_k' \theta^{(j)} - y_k - \delta & \text{if } \varphi_k' \theta^{(j)} > y_k + \delta \\ y_k - \varphi_k' \theta^{(j)} - \delta & \text{if } \varphi_k' \theta^{(j)} < y_k - \delta \\ 0 & \text{otherwise} . \end{cases}$$

The basic idea is to favor updates of the current estimate $\theta^{(j)}$ which aim at correcting unsatisfied inequalities with a relatively small violation. Indeed, the

[2] The algorithm is illustrated in its first application to the overall system (5), but it can be easily specialized when, in subsequent applications, only a subsystem of system (5) is considered.

correction of an unsatisfied inequality with large violation is likely to corrupt other inequalities that $\theta^{(j)}$ satisfies. A decreasing temperature parameter T, which the violations are compared with, is therefore introduced in order to give decreasing attention to unsatisfied inequalities with large violations. The algorithm can be formalized as follows.

```
Given: $C$, $T_0$, $\theta^{(1)}$;
Set $c = 0$, $j = 1$, $\bar{\theta} = \theta^{(1)}$;
while $c < C$ do
Initialize the set of indices $I = \{1, \dots, N\}$ and set $T = (1 - c/C)T_0$;
repeat
    Pick the index $k$ from $I$ according to the prescribed rule;
    Compute the violation $v_j^k$ and set $\lambda_j = (T/T_0)\exp(-v_j^k/T)$;
    if $\varphi_k'\theta^{(j)} > y_k + \delta$ then $\theta^{(j+1)} = \theta^{(j)} - \lambda_j\varphi_k$;
    else if $\varphi_k'\theta^{(j)} < y_k - \delta$ then $\theta^{(j+1)} = \theta^{(j)} + \lambda_j\varphi_k$;
    else $\theta^{(j+1)} = \theta^{(j)}$;
    if $\theta^{(j+1)} \neq \theta^{(j)}$ and $\theta^{(j+1)}$ satisfies a larger number of complementary
    inequalities than $\bar{\theta}$ then $\bar{\theta} = \theta^{(j+1)}$;
    Set $I = I - \{k\}$ and $j = j + 1$;
until $I = \emptyset$
Set $c = c + 1$;
end.
```

All the complementary inequalities satisfied by $\bar{\theta}$ form a feasible subsystem of system (5), which is the solution of the extended MAX FS returned by the algorithm. Notice that this solution is not guaranteed to be optimal, even though $\bar{\theta}$ is the estimate that, during the process, has satisfied the largest number of complementary inequalities.

For the choice of C and T_0, as well as for practical questions concerning the implementation of the algorithm, we refer to [8].

3.2 Comments about the Greedy Approach to MIN PFS

Let us denote by $\hat{s}$ the number of feasible subsystems of system (5) provided by successive applications of the algorithm described in Section 3.1. The estimate of the number of affine submodels needed to fit the data and the classification of the datapoints thus provided suffer two drawbacks. First, it is not guaranteed to yield minimum partitions, *i.e.*, the number of submodels $\hat{s}$ could be larger than the minimum number s needed, *e.g.*, because two subsets of complementary inequalities that could be satisfied by one and the same parameter vector, are extracted at two different iterations. Second, since some datapoints might be consistent with more than one submodel, the cardinality and the composition of the clusters could depend on the order in which the close-to-maximum feasible subsystems are extracted.

In order to cope with these drawbacks, a procedure for the refinement of the estimates will be proposed in the next section. As we will show, such a

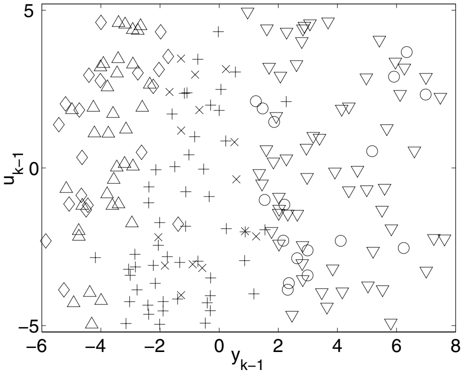

Fig. 2. Initial classification of the regression vectors. Each mark corresponds to a different cluster, for a total of six clusters

procedure improves both the classification of the datapoints and the quality of the fit by properly reassigning the datapoints and selecting pointwise estimates of the parameter vectors that characterize each submodel.

Notice that one could decide to stop the algorithm when the cardinalities of the extracted clusters become too small. This might be useful in order to penalize submodels that account for just a few datapoints (that, most likely, are outliers).

*Example 1 (*cont'd*).* We ran the greedy algorithm (with $C = 100$, $T_0 = 100$ and cyclic selection of the datapoints) over the set of datapoints of Example 1. Since the noise was uniformly distributed in $[-0.1, 0.1]$, the bound δ was chosen equal to 0.1 accordingly. We found a complete partition into $\hat{s} = 6$ clusters, containing 52, 61, 35, 17, 20 and 15 datapoints, respectively. The six clusters of regression vectors are depicted in Figure 2. The number of submodels is overestimated, and from the comparison of Figures 1 and 2 it is evident that regression vectors belonging to the same region were extracted at different iterations of the algorithm.

4 Refinement of the Estimates

The initialization of the identification procedure described in Section 3 provides the clusters $\mathcal{D}_i^{(0)}$ which consist of all the datapoints (y_k, x_k) corresponding to the i-th extracted feasible subsystem of system (5), $i = 1, \ldots, \hat{s}$. Moreover, each feasible subsystem defines the set of feasible parameter vectors for the corresponding affine submodel.

As discussed in Section 3.2, a refinement procedure is required in order to improve both the classification of the datapoints and the quality of the fit. The basic procedure that we propose consists of two main steps to be iterated. In the first step, all the datapoints are classified according to the current estimated parameter vectors. In the second step, new pointwise estimates of the parameter vectors are selected on the basis of the previously computed clusters of datapoints. This is performed by using the *projection estimate* defined as

$$\Phi_p(\mathcal{D}) = \arg\min_{\theta} \max_{(y_k,x_k)\in\mathcal{D}} \left| y_k - \varphi_k'\theta \right| , \tag{6}$$

where $\mathcal{D}$ is a cluster of datapoints (y_k, x_k). Notice that the computation of the projection estimate can be formulated as a suitable *linear programming* (LP) problem. The refinement procedure can be formalized as follows.

0. **Initialization**
   Set $t = 1$ and select a termination threshold $\gamma \geq 0$.
   For $i = 1, \dots, \hat{s}$, set $\hat{\theta}_i^{(1)} = \Phi_p(\mathcal{D}_i^{(0)})$.
1. **Reassignment of the datapoints**
   For each datapoint (y_k, x_k), $k = 1, \dots, N$:
   - If $\left|y_k - \varphi_k'\hat{\theta}_i^{(t)}\right| > \delta$ for all $i = 1, \dots, \hat{s}$, then mark (y_k, x_k) as *infeasible.*
   - If $\left|y_k - \varphi_k'\hat{\theta}_i^{(t)}\right| \leq \delta$ for more than one $i = 1, \dots, \hat{s}$, then mark (y_k, x_k) as *undecidable.*
   - If $\left|y_k - \varphi_k'\hat{\theta}_i^{(t)}\right| \leq \delta$ for only one $i = 1, \dots, \hat{s}$, then assign (y_k, x_k) to $\mathcal{D}_i^{(t)}$ and mark it as *feasible.*
2. **Re-estimation of the parameter vectors**
   For $i = 1, \dots, \hat{s}$, compute $\hat{\theta}_i^{(t+1)} = \Phi_p(\mathcal{D}_i^{(t)})$.
3. **Termination**
   If $\|\hat{\theta}_i^{(t+1)} - \hat{\theta}_i^{(t)}\|/\|\hat{\theta}_i^{(t)}\| \leq \gamma$ for all $i = 1, \dots, \hat{s}$, then exit. Otherwise, set $t = t + 1$ and go to step 1.

In order to avoid that the procedure does not terminate, only a maximum number $t_{\max}$ of refinements is allowed. Convergence properties of the procedure are currently under investigation.

The basic idea of the procedure is that, while the new parameter vectors $\hat{\theta}_i^{(t+1)}$ are computed on the basis of the clusters $\mathcal{D}_i^{(t)}$, some infeasible, as well as undecidable, datapoints may become feasible, *i.e.*, may be assigned to some cluster $\mathcal{D}_i^{(t+1)}$, thus improving the quality of the classification. Notice that the use of the projection estimate in step 2 guarantees that no feasible datapoint at refinement t becomes infeasible at refinement $t + 1$, since

$$\max_{(y_k,x_k)\in\mathcal{D}_i^{(t)}} \left|y_k - \varphi_k'\hat{\theta}_i^{(t+1)}\right| \leq \max_{(y_k,x_k)\in\mathcal{D}_i^{(t)}} \left|y_k - \varphi_k'\hat{\theta}_i^{(t)}\right| \leq \delta \ , \quad i = 1, \dots, \hat{s} \ .$$

In step 1 the distinction among infeasible, undecidable, and feasible datapoints is motivated by the following considerations. If the estimated parameter vectors provide a good fit of the data, it is likely that a datapoint (y_k, x_k) considerably violating the inequalities $\left|y_k - \varphi_k'\hat{\theta}_i^{(t)}\right| \leq \delta$, $i = 1, \dots, \hat{s}$, is an outlier. Hence, it is reasonable to expect that neglecting the infeasible datapoints in the parameter re-estimation helps to improve the quality of the fit. The undecidable datapoints are instead consistent with more than one submodel. This indecision (that is inherent with the data) could be solved only by exploiting the partition of the PWA map. In this phase, neglecting the undecidable datapoints helps to reduce the number of misclassifications. As it will be clarified in the next section, this will make possible a better estimation of the PWA partition.

The basic procedure for the refinement of the estimates does not change the estimated number of submodels, so that further steps are required to cope with the case when the greedy algorithm provides an overestimation of the number of submodels needed to fit the data. Recall that this could occur because of the suboptimality of the greedy strategy, and the randomness of the method used to tackle the extended MAX FS problem.

In order to decrease the number of submodels, we can use information about the estimated parameter vectors and the cardinalities of the clusters. In fact, if two subsets of complementary inequalities can be satisfied by one and the same parameter vector, it is likely that the corresponding estimated parameter vectors are very similar (and we possibly have a large number of undecidable datapoints), so that they can be merged into one subset. On the other hand, if during the refinement of the estimates the cardinality of a cluster becomes too small with respect to N, the corresponding submodel can be discarded, since it accounts only for few datapoints (most likely outliers). Additional steps to the basic procedure are thus the following (α and β are fixed nonnegative thresholds):

- **Similarity of the parameter vectors**
  Compute $\alpha_{i^*,j^*} = \min_{1 \leq i < j \leq \hat{s}} \|\hat{\theta}_i^{(t)} - \hat{\theta}_j^{(t)}\| / \min\left\{\|\hat{\theta}_i^{(t)}\|, \|\hat{\theta}_j^{(t)}\|\right\}$. If $\alpha_{i^*,j^*} \leq \alpha$, merge the submodels i^* and j^*, update the number of submodels $\hat{s}$, and renumber the submodels from 1 to $\hat{s}$.
- **Cardinality of the clusters**
  Compute $\beta_{i^*} = \min_{i=1,\ldots,\hat{s}} \dim(\mathcal{D}_i^{(t)})/N$. If $\beta_{i^*} \leq \beta$, discard the i^*-th submodel, update the number of submodels $\hat{s}$, and renumber the remaining submodels (and, accordingly, the corresponding clusters) from 1 to $\hat{s}$. Then, reassign only the undecidable datapoints as in step 1.

The similarity of the parameter vectors is to be tested before the reassignment of the datapoints. The fusion of two submodels i^* and j^* can be performed in different ways. For instance, the fused parameter vector can be computed as the mean $\left(\hat{\theta}_{i^*}^{(t)} + \hat{\theta}_{j^*}^{(t)}\right)/2$, or on the basis of the union of the clusters $\mathcal{D}_{i^*}^{(t-1)}$ and $\mathcal{D}_{j^*}^{(t-1)}$, using the projection estimate (6). This latter computation generally provides better performance. The cardinality of the clusters is instead to be tested after the reassignment of the datapoints.

The thresholds α and β should be suitably chosen in order to decrease the number of submodels still preserving a good fit of the data. Indeed, it is clear that, if such thresholds are chosen too large, the number of submodels might decrease under s. In this case, the number of infeasible datapoints increases, since some significant dynamics is no more in the model. One could use this information to adjust α and β, and then repeat the refinement. In general, when the procedure terminates, the number of infeasible datapoints is always an index of the quality of the fit.

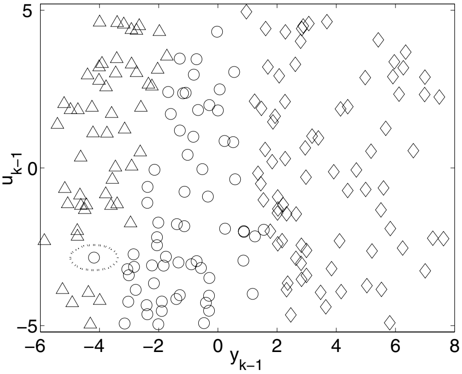

Fig. 3. Classification of the regression vectors (*triangles, circles, diamonds*) after the refinement

*Example 1 (*cont'd*).* We performed the refinement procedure with $\alpha = 15\%$, $\beta = 1\%$ and $\gamma = 0.001\%$. The termination condition was reached after six refinements. The number $\hat{s}$ of estimated submodels decreased from 6 to 3. The corresponding three clusters of regression vectors are depicted in Figure 3, and contain 53, 65 and 79 points, respectively. Two datapoints are left infeasible, and only one is undecidable. The finally estimated parameter vectors are

$$\hat{\theta}_1 = \begin{bmatrix} -0.3921 \\ 0.9978 \\ 1.5426 \end{bmatrix}, \quad \hat{\theta}_2 = \begin{bmatrix} 0.4980 \\ -0.9994 \\ -0.4971 \end{bmatrix}, \quad \hat{\theta}_3 = \begin{bmatrix} -0.3000 \\ 0.5005 \\ -1.7011 \end{bmatrix},$$

providing very good estimates of the true submodels. In Figure 3 the highlighted (misclassified) circle shows that the clusters marked with triangles and circles are not linearly separable. It is worth noticing in Figure 4 how the number of undecidable datapoints considerably decreases as the number of refinements increases, *i.e.*, as the number of estimated submodels is reduced.

5 Estimation of the Partition of the Regressor Set

So far we have classified the datapoints and estimated the affine submodels. The final step of the identification procedure consists in estimating the partition of the regressor set. This step can be performed by considering pairwise the clusters $\mathcal{F}_i = \{x_k | (y_k, x_k) \in \mathcal{D}_i\}$ (where $\mathcal{D}_i$, $i = 1, \ldots, \hat{s}$, is the final classification of the feasible datapoints provided by the refinement procedure), and finding a separating hyperplane for each of such pairs.

Given two linearly separable clusters $\mathcal{F}_i$ and $\mathcal{F}_j$, with $i \neq j$, a *separating hyperplane* $x'a + b = 0$, with $a \in \mathbb{R}^n$ and $b \in \mathbb{R}$, is such that, for some $\varepsilon > 0$,

$$\begin{cases} x_k' a + b \leq -\varepsilon & \forall x_k \in \mathcal{F}_i \\ x_k' a + b \geq \varepsilon & \forall x_k \in \mathcal{F}_j \,. \end{cases} \tag{7}$$

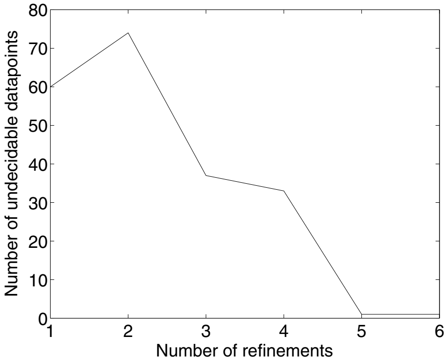

Fig. 4. Number of undecidable datapoints vs the number of refinements

If the two clusters $\mathcal{F}_i$ and $\mathcal{F}_j$ are not linearly separable, a hyperplane that minimizes the number of misclassified points (*i.e.*, points x_k not satisfying (7)) is called *generalized separating hyperplane*. Notice that, even though the true function $F(\cdot)$ were a PWA map defined over a polyhedral partition of the $\mathcal{X}$-domain, two clusters $\mathcal{F}_i$ and $\mathcal{F}_j$ might anyway not be linearly separable due to classification errors. This kind of errors is actually expected to be reduced by the distinction into *infeasible*, *undecidable*, and *feasible* datapoints. Indeed, the infeasible datapoints account mainly for the outliers, whereas the undecidable datapoints are those that most likely could induce misclassifications, since they are consistent with more than one submodel.

Linear Support Vector Machines (SVMs) are a suitable tool for this stage of the identification procedure, since they accomplish simultaneously the distinct tasks of finding the *optimal separating hyperplane* of two clusters of points (*i.e.*, the separating hyperplane that maximizes the distance from the closest point of each cluster), while minimizing the number of misclassified points [11].

In this paper, we show that the problem of finding a generalized separating hyperplane (thus providing also linearly separable clusters of points) can be formulated as a MAX FS problem. Indeed, given two clusters $\mathcal{F}_i$ and $\mathcal{F}_j$, according to (7) a separating hyperplane turns out to be a solution of the system of linear inequalities $\Phi \begin{bmatrix} a \\ b \end{bmatrix} \leq \xi$, where the rows of Φ are the vectors φ'_k for all $x_k \in \mathcal{F}_i$ and $-\varphi'_k$ for all $x_k \in \mathcal{F}_j$, and ξ is a column vector of $-\varepsilon$'s. If such system is infeasible, solving a MAX FS problem clearly corresponds to finding a hyperplane that minimizes the number of misclassified points. The misclassified points, if any, are then removed from $\mathcal{F}_i$ and/or $\mathcal{F}_j$. Since MAX FS is NP-hard, the randomized relaxation method for MAX FS proposed in [12] (of which the algorithm presented in Section 3.1 is a straightforward estension) can be used to provide good solutions in a short amount of computation time.

Optionally, once two linearly separable clusters of points are available, one could look for the optimal separating hyperplane of the two clusters. As detailed in [11], this can be performed by solving a *quadratic programming* (QP) problem:

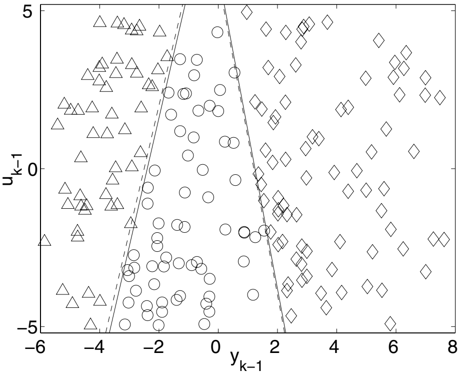

Fig. 5. Final classification of the regression vectors (*triangles, circles, diamonds*), and true (*dashed lines*) and estimated (*solid lines*) partition of the regressor set

$$\begin{aligned} &\min_{(a,b)} \frac{1}{2}\|a\|^2 \\ &\text{subject to} \quad \begin{cases} x_k'a + b \le -1 & \forall x_k \in \mathcal{F}_i \\ x_k'a + b \ge 1 & \forall x_k \in \mathcal{F}_j \ . \end{cases} \end{aligned} \tag{8}$$

This separating hyperplane is termed *optimal* as it separates the two clusters with the maximal margin[3]. Each estimated region $\hat{\mathcal{X}}_i$, $i = 1, \dots, \hat{s}$, is then defined by all the optimal hyperplanes separating $\mathcal{F}_i$ from $\mathcal{F}_j$, with $j \neq i$.

The method for the estimation of the PWA partition based on separating hyperplanes has two major drawbacks. First, if the two clusters $\mathcal{F}_i$ and $\mathcal{F}_j$ are not contiguous, the corresponding separating hyperplane possibly does not contribute to delimiting the estimated regions $\hat{\mathcal{X}}_i$ and $\hat{\mathcal{X}}_j$, *i.e.*, we have redundancy in the representation of the partition. Second, when $n > 1$, this method does not guarantee that the estimated regions form a complete partition of the regressor set.

The former drawback can be overcome by eliminating redundant hyperplanes through standard linear programming techniques. The latter drawback is more important, since it causes the model to be not completely defined over the whole regressor set. However, both in simulation and optimization the presence of "holes" in the PWA partition can be often accepted. In simulation, when a regression vector falls into a "hole", it can be reasonably assigned to the nearest region, whereas for optimization purposes, trajectories passing through a "hole" are simply automatically discarded as infeasible, with the only consequent drawback of inducing suboptimal solutions. We are currently investigating how to avoid the presence of such "holes" by partitioning them into additional convex sets.

[3] It can be easily shown that, in (8), at least one "≤" and one "≥" constraint are active at the optimum, so that the distance of the closest point of each cluster from the optimal hyperplane is $1/\|a\|$.

The above mentioned techniques generally provide satisfactory results when the number of misclassified points is small with respect to the cardinalities of the two clusters $\mathcal{F}_i$ and $\mathcal{F}_j$. If this is not the case, at least one of $\mathcal{F}_i$ and $\mathcal{F}_j$ needs to be partitioned. Notice that, when a cluster $\mathcal{F}_i$ must be partitioned, this may correspond to nonconnected regions where the parameter vector is the same (recall that the classification procedure groups together the datapoints only according to the fact that they are fitted by the same affine submodel), or to a nonconvex region that needs to be split into convex polyhedra. Techniques for partitioning a cluster $\mathcal{F}_i$ by exploiting the information about the misclassified points while separating $\mathcal{F}_i$ from $\mathcal{F}_j$, with $j \neq i$, are currently under investigation.

*Example 1 (*cont'd*).* The final classification of the regression vectors and the estimated partition of the regressor set are depicted in Figure 5. The line separating triangles and diamonds has not been drawn, since it is redundant, whereas the two solid lines are defined by the coefficient vectors

$$\hat{h}_1 = \left[4.0036\ {-0.9854}\ 9.5903\right]' \ , \quad \hat{h}_2 = \left[5.0002\ 0.9990\ {-6.2009}\right]' \ ,$$

that are very similar to the true ones. Notice that in Figure 3 the clusters marked with triangles and circles are not linearly separable (consider the highlighted circle), so that the pre-separation exploiting MAX FS is actually useful to detect the misclassification and to provide linearly separable clusters. The overall computation of the estimated PWA model took about 7 seconds on an AMD Athlon 1GHz running Matlab 6.1 non-optimized code.

Example 2. The PWA identification algorithm was successfully applied to fit the data generated by a discontinuous PWARX system for which the regression vector was $x_k = [y_{k-1}\ y_{k-2}\ u_{k-1}\ u_{k-2}]'$ (so that $n = 4$), and $s = 4$. The input signal u_k was chosen to be uniformly distributed in $[-5, 5]$, and the noise signal e_k was assumed to be normally distributed with zero mean and variance $\sigma^2 = 0.2$. δ was chosen equal to $3\sigma = 1.34$. $N = 1000$ and $N_{\mathrm{V}} = 500$ datapoints were used for estimation and validation, respectively. The algorithm provided $\hat{s} = s = 4$ submodels. The true and the estimated parameter vectors are shown in Table 1. The validation of the model was performed by computing the prediction error, *i.e.*, the difference between the measured and the predicted output, whose plot is depicted in Figure 6. Notice that it is mostly contained between δ and $-\delta$. Spikes

Table 1. True and estimated parameter vectors for Example 2

θ_1	$\hat{\theta}_1$	θ_2	$\hat{\theta}_2$	θ_3	$\hat{\theta}_3$	θ_4	$\hat{\theta}_4$
-0.05	-0.09	1.21	1.22	1.49	1.48	-1.20	-1.25
0.76	0.77	-0.49	-0.50	-0.50	-0.52	-0.72	-0.65
1.00	1.04	-0.30	-0.28	0.20	0.23	0.60	0.65
0.50	0.45	0.90	0.89	-0.45	-0.36	-0.70	-0.80
0	0.08	0	-0.13	0	0.20	0	0.37

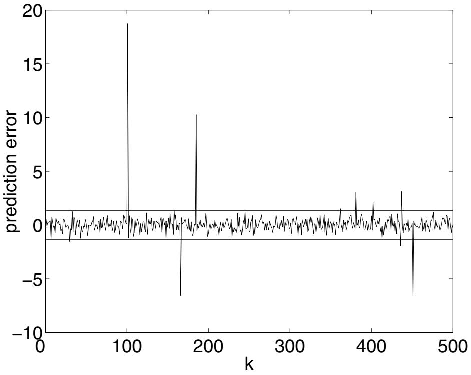

Fig. 6. Plot of the prediction error for Example 2

are due to regression vectors assigned to the wrong submodel because of errors in the estimation of the PWA partition, and to discontinuity of the PWA map. The overall computation of the estimated PWA model took about one minute and half on an AMD Athlon 1GHz running Matlab 6.1 non-optimized code. Notice that this example is quite challenging, due to the quite low signal/noise ratio and the high number of parameters to be estimated with respect to the available data.

6 Conclusions

In this paper we considered the problem of identifying a PWA model of a (possibly non-smooth) discrete-time nonlinear system from input-output data. We proposed a two-stage procedure that first divides the data into clusters and estimates the parameters of the affine submodels, and then estimates the coefficients of the hyperplanes defining the partition of the PWA map.

In order to provide an initial clustering of the datapoints, we adopted the greedy strategy proposed in [8]. The major capability of this strategy is that it also provides an estimate of the number of submodels needed to fit the data. Other approaches could be used to initialize the identification procedure (*e.g.*, the k-plane clustering algorithm proposed in [15]). Then, we proposed an algorithm for improving both the classification of the datapoints and the estimation of the parameters. The algorithm alternates between datapoint reassignment and parameter update. Moreover, the number of submodels is allowed to vary from iteration to iteration. This is made possible by introducing the thresholds α and β, which the similarities of the parameter vectors and the cardinalities of the clusters of datapoints are compared with, respectively. Current research is aimed at deriving rules for the automatic selection and update of α and β, in order to completely automatize the algorithm and to further improve its performance, and at investigating convergence properties of the algorithm.

The partition of the PWA map is finally estimated by considering pairwise the clusters of regression vectors, and finding a separating hyperplane for each of such pairs. We are also currently investigating how to avoid the presence of "holes" in the resulting partition, and how to split the clusters corresponding to nonconvex regions, or to nonconnected regions where the affine submodel is the same.

References

1. Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P., Hjalmarsson, H., Juditsky, A.: Nonlinear black-box modeling in system identification: a unified overview. Automatica **31** (1995) 1691–1724
2. Juditsky, A., Hjalmarsson, H., Benveniste, A., Delyon, B., Ljung, L., Sjöberg, J., Zhang, Q.: Nonlinear black-box models in system identification: mathematical foundations. Automatica **31** (1995) 1725–1750
3. Bemporad, A., Ferrari-Trecate, G., Morari, M.: Observability and controllability of piecewise affine and hybrid systems. IEEE Trans. Automatic Control **45** (2000) 1864–1876
4. Heemels, W., Schutter, B.D., Bemporad, A.: Equivalence of hybrid dynamical models. Automatica **37** (2001) 1085–1091
5. Ferrari-Trecate, G., Muselli, M., Liberati, D., Morari, M.: A clustering technique for the identification of piecewise affine systems. Automatica **39** (2003) 205–217
6. Bemporad, A., Roll, J., Ljung, L.: Identification of hybrid systems via mixed-integer programming. In: Proc. 40th IEEE Conf. on Decision and Control. (2001) 786–792
7. Münz, E., Krebs, V.: Identification of hybrid systems using apriori knowledge. In: Proc. 15th IFAC World Congress. (2002)
8. Amaldi, E., Mattavelli, M.: The MIN PFS problem and piecewise linear model estimation. Discrete Applied Mathematics **118** (2002) 115–143
9. Milanese, M., Vicino, A.: Optimal estimation theory for dynamic systems with set membership uncertainty: an overview. Automatica **27** (1991) 997–1009
10. Milanese, M., Norton, J.P., Piet-Lahanier, H., (eds.), E.W.: Bounding Approaches to System Identification. Plenum Press, New York (1996)
11. Vapnik, V.: Statistical Learning Theory. John Wiley (1998)
12. Amaldi, E., Hauser, R.: Randomized relaxation methods for the maximum feasible subsystem problem. Technical Report 2001-90, DEI, Politecnico di Milano, Italy (2001)
13. Agmon, S.: The relaxation method for linear inequalities. Canadian J. Math. **6** (1954) 382–392
14. Motzkin, T., Schoenberg, I.: The relaxation method for linear inequalities. Canadian J. Math. **6** (1954) 393–404
15. Bradley, P.S., Mangasarian, O.L.: k-plane clustering. Journal of Global Optimization **16** (2000) 23–32