# A new hybrid system identification algorithm with automatic tuning

**Fabien Lauer and Gérard Bloch**

*Centre de Recherche en Automatique de Nancy (CRAN),*
*Nancy–University, CNRS, France,*
*fabien.lauer@esstin.uhp-nancy.fr*
*gerard.bloch@esstin.uhp-nancy.fr*

**Abstract:** Hybrid system identification is composed of two subproblems: estimate the discrete state or mode for each data point, and estimate the submodel governing the dynamics of the continuous state for each mode. For linear hybrid systems, the paper proposes to tackle these problems in a single step by simultaneously approximating each submodel while associating data points to each of these. The method borrows ideas from bounded-error approaches and Support Vector Regression to extend the algebraic procedure. The algorithm can easily deal with noise by fixing a predefined accuracy threshold. This bound on the error can be different for each mode when the noise level is considered to switch with the system. An extension of the algorithm to automatically tune itself in accordance with the noise level of each mode is also provided. The method can be seen as an an extension of the algebraic approach, but also as an alternative to the bounded error approach when a predefined or preferred model structure is given or when the noise level is unknown. An extension of the method to nonlinear submodels is provided in the companion paper (Lauer and Bloch, 2008).

Keywords: hybrid system; system identification; bounded error; Support Vector Regression.

## 1. INTRODUCTION

*Context.* A hybrid system is usually described by both a continuous state and a discrete state, where the vector field defining the evolution of the continuous state depends on the discrete state. In this framework, a system can be seen as switching between different subsystems, which are usually modeled by linear AutoRegressive with eXogenous inputs (ARX) models in the discrete-time case. Two types of identification problems may arise in this setting depending on whether the discrete state sequence that generated the data is known or not. If it is, then the problem can be simply recast as $n$ common identification problems, each one using only the data for a given discrete state. However, in most cases this sequence is unknown and the problem becomes nontrivial as it amounts to both estimate the regions or the boundaries in the continuous state space (or regressor space) that define the discrete state and the parameters of the linear models in these regions.

*Related work.* Five main approaches have been devised for this problem: the clustering-based approach (Ferrari-Trecate et al., 2003), the mixed integer programming based approach (Roll et al., 2004), the Bayesian approach (Juloski et al., 2005a), the bounded error approach (Bemporad et al., 2005) and the algebraic approach (Vidal et al., 2003). The first four focus on the problem of PieceWise Affine (PWA) system identification, where the discrete state depends on the continuous state. However, both the bounded error and Bayesian approaches can also be used to identify a broader class of systems, known as switched linear systems, where the discrete state evolves independently of the continuous state. The algebraic approach (Vi-

dal et al., 2003) focuses on this latter problem, but without taking the noise into account in its development. This leads to an algorithm very sensitive to noise, compared to the clustering-based or bounded error methods, as shown by Juloski et al. (2005b). Besides, the bounded error method provides a convenient way of dealing with noisy data by looking for a model with a predefined accuracy. However, the hyperparameters of the method, such as the model accuracy that determines the number of modes, may be difficult to tune to get a prescribed structure, e.g. if prior knowledge on the number of modes is available (Juloski et al., 2005b).

*Tools and proposed method.* The paper focuses on linear hybrid system identification [1] from noisy data with Switched ARX (SARX) models. PieceWise ARX (PWARX) models are considered as a special case of this problem. In this context, the paper proposes a bounded error approach to satisfy the hybrid decoupling constraint when working with noisy data in the algebraic identification framework. The new method uses tools and ideas from the fields of machine learning and optimization. In machine learning, Support Vector Machines (SVM) (Vapnik, 1995) can be used both for regression and system identification (Drezet and Harrison, 1998; Mattera and Haykin, 1999). In hybrid system identification, SVM classifiers are already used to estimate the switching boundaries between the modes (Ferrari-Trecate et al., 2003; Bemporad et al., 2005). For regression, this learning algorithm, known as Support Vector Regression (SVR), uses an $\varepsilon$-insensitive

---

[1] Nonlinear hybrid system identification is discussed in the companion paper (Lauer and Bloch, 2008).

loss function which does not take into account errors that are less than $\varepsilon$ (Smola and Schölkopf, 2004). This loss function allowing for errors below a predefined threshold is close in spirit to the bounded error approach. However, the origin is different. In learning theory, this effect is justified in order to minimize the generalization error of the model, whereas the bounded error approach was developed to allow the automatic determination of the number of linear submodels required to approximate a nonlinear function with a given accuracy.

*Contribution.* The paper proposes a new method for linear hybrid system identification from noisy data. This method allows to consider a noise model that also switches with the system by choosing different bounds on the error for different modes. To our knowledge, this problem has not yet been explored in the literature, though being realistic (most sensors have a constant relative precision rather than a constant absolute one). An extension of the algorithm to automatically tune itself in accordance with the noise level of each mode is also provided. In comparison to the the bounded error method (Bemporad et al., 2005), which determines the number of modes for a fixed bound on the error, this alternative approach to the tuning of parameters may be useful when a predefined or preferred model structure is given. In this case, instead of testing various thresholds on the accuracy until a good model structure is found, a minimal bound on the error can be automatically determined for a given number of modes.

*Paper organization.* Starting from the setting of the problem in the algebraic framework (sect. 2.1), the paper then describes how to deal with noise in a bounded error approach (sect. 2.2). Then, the bounded error with soft constraints is discussed with the use of the $\varepsilon$-insensitive loss function (sect. 2.3) before exposing the algorithm itself (sect. 2.4). This alorithm is then extended to automatically tune the bounds on the error (sect. 2.5). Finally, two numerical examples are given (sect. 3) to show both the automatic tuning and the application to hybrid system identification.

Notations: all vectors are column vectors written in boldface and lowercase letters whereas matrices are written in boldface and uppercase.

## 2. HYBRID SYSTEM IDENTIFICATION

Consider modeling a linear hybrid system of output $y_i$ from $N$ input-output samples $(\boldsymbol{x}_i, y_i)$ by a model $f$,

$$y_i = f(\boldsymbol{x}_i) + e_i = f_{\lambda_i}(\boldsymbol{x}_i) + e_i, \qquad (1)$$

composed of $n$ ARX affine [2] submodels $f_j$ of the form

$$f_j(\boldsymbol{x}_i) = \boldsymbol{w}_j^T \boldsymbol{x}_i + b_j, \qquad (2)$$

where $\boldsymbol{w}_j \in \mathbb{R}^p$ and $b_j \in \mathbb{R}$ are the $p+1$ parameters, the regression vector $\boldsymbol{x}_i = [y_{i-1} \ \ldots \ y_{i-n_a}, \ u_{i-1} \ \ldots \ u_{i-n_c}]^T$ contains the delayed $n_a$ outputs $y_{i-k}$ and $n_c$ inputs $u_{i-k}$ of the system, while the discrete state $\lambda_i \in \{1, 2, \ldots, n\}$ determines which mode (or which ARX submodel) is used to compute $f(\boldsymbol{x}_i)$. In a Switched ARX (SARX) model, the discrete state $\lambda_i$ evolves independently of $\boldsymbol{x}_i$, whereas in a

PieceWise ARX (PWARX) model, the discrete state is a function of $\boldsymbol{x}_i$. This function is defined as

$$\lambda_i = j, \text{ if } \boldsymbol{x}_i \in S_j = \{\boldsymbol{x} \ : \ \boldsymbol{H}_j \begin{bmatrix} \boldsymbol{x}_i \\ 1 \end{bmatrix} \leq \boldsymbol{0}\}, \ j = 1, \ldots, n,$$
$$(3)$$

in which the matrices $\boldsymbol{H}_j$ define the regions $S_j$ partitioning the regressor space $\mathcal{X} \subseteq \mathbb{R}^{(n_a+n_c)}$.

### 2.1 Noiseless case: the algebraic approach

Vidal et al. (2003) proposed an algebraic approach to simultaneously estimate the discrete and the continuous parts of a SARX model. It basically amounts to consider that the vector $\boldsymbol{z}_i = [\boldsymbol{x}_i^T, \ -y_i]^T$ lies at least on one of the hyperplanes defined by [3] $[\boldsymbol{w}_j^T \ 1]\boldsymbol{z}_i + b_j = 0$, i.e. that the product, $\prod_{j=1}^n [\boldsymbol{w}_j^T \ 1]\boldsymbol{z}_i + b_j$, of these terms, known as the hybrid "decoupling" polynomial, equals zero. The solution is then obtained by writing this constraint for all the $N$ available data points $(\boldsymbol{x}_i, y_i)$ and developing the polynomials to recover a linear system of equations involving new parameters $\boldsymbol{h}_j$ built from products of the original parameters $\boldsymbol{w}_j, b_j$.

This method can be seen as forcing at least one submodel $f_j$ to approximate the point $(\boldsymbol{x}_i, y_i)$ with zero error. Defining the submodel errors $e_{ij} = y_i - f_j(\boldsymbol{x}_i) = y_i - \boldsymbol{w}_j^T \boldsymbol{x}_i - b_j$, and imposing this condition for the $N$ points allows to write the hybrid "decoupling" polynomial constraints as

$$\prod_{j=1}^n e_{ij} = 0, \ i = 1, \ldots, N. \qquad (4)$$

However, in the case of noisy data, these constraints cannot be satisfied. In practice, Vidal et al. (2003) proposes to solve the linear system involving the products $\boldsymbol{h}_j$ of the parameters in a least squares sense to approximate the solution. As detailed by Ma and Vidal (2005), this corresponds to the minimization of a sub-optimal approximation, suitable for a low noise level, of the objective function $\sum_{i=1}^N e_i^2$:

$$\min_{\boldsymbol{h}_j} \sum_{i=1}^N \prod_{j=1}^n (e_{ij})^2. \qquad (5)$$

One drawback of this method is that the original parameters $\boldsymbol{w}_j$ and $b_j$ are estimated through their products and not directly, which may lead to additional errors on their estimates.

### 2.2 Considering noise with a bounded error approach

The bounded error approach developed by Bemporad et al. (2005) aims at finding the minimal number $n$ of submodels that allows the error $e_i$ on all the training points $(\boldsymbol{x}_i, y_i)$ to be bounded by

$$|y_i - f(\boldsymbol{x}_i)| = |e_i| \leq \delta, \ i = 1, \ldots, N. \qquad (6)$$

Considering this approach in the setting of section 2.1 results in rewriting the "decoupling" polynomials (4) as

$$\prod_{j=1}^n [|e_{ij}| \geq \delta] = 0, \ i = 1, \ldots, N, \qquad (7)$$

---

[2] Affine models are more general than linear ones and allow the piecewise function $f$ to be continuous in PWARX models.

[3] In Vidal et al. (2003), only linear submodels without the term $b_j$ are considered.

where $[\cdot] = 1$, if the bracketed expression is true, and 0 otherwise. If the discrete state is estimated by

$$\hat{\lambda}_i = \arg\min_{j \in \{1,\dots,n\}} |e_{ij}|, \qquad (8)$$

then the absolute value of the error of the hybrid model, $|e_i| = \min_j |e_{ij}|$, is bounded by the threshold $\delta$, for all $i$ for which constraint (7) holds. However, the problem thus posed is not easy to solve. The following proposes a relaxation of the bounded error constraint through the use of a particular loss function.

### 2.3 Bounded error with soft constraints

The $\varepsilon$-insensitive loss function, defined by Vapnik (1995) for Support Vector Regression (SVR) as

$$l(e_{ij}) = \begin{cases} 0 & \text{if } |e_{ij}| \leq \varepsilon, \\ |e_{ij}| - \varepsilon & \text{otherwise,} \end{cases} \qquad (9)$$

builds a tube of insensitivity, in which the errors are meaningless. Absolute errors larger than the width of the tube $\varepsilon$ are penalized linearly, which ensures a certain robustness to outliers compared to a quadratic loss function.

The bounded error constraints (7) can thus be approximated efficiently by the minimization of this loss function. In this setting, the problem becomes

$$\min_{\boldsymbol{w}_j, b_j} \sum_{i=1}^{N} \prod_{j=1}^{n} l(e_{ij}). \qquad (10)$$

**Remark 1:** *Compared to the approach of Bemporad et al. (2005), solving (10) does not ensure that the error is truly bounded for all points. However, this softening of the constraints can be desirable to deal with outliers.*

**Remark 2:** *Compared to the approach of Vidal et al. (2003), the objective function in (10) is minimized directly with respect to the model parameters $\boldsymbol{w}_j$ and $b_j$ instead of their products. This may prevent additional errors on the parameter estimates.*

An issue that has not been studied in the literature so far is the identification of hybrid systems under the assumption of a switching noise model. This framework extends the standard models in order to consider that the different submodels are corrupted by noise of different levels and or distributions.

An advantage of the proposed approach is the possibility to deal easily with a different noise level for each mode by using different bounds on the errors. Simply define $n$ loss functions $l_j$ as

$$l_j(e_{ij}) = \begin{cases} 0 & \text{if } |e_{ij}| \leq \varepsilon_j, \\ |e_{ij}| - \varepsilon_j & \text{otherwise,} \end{cases} \qquad (11)$$

where $\varepsilon_j$ can be set to different values for each $j$. Then these loss functions can be used in (10) to accomodate each mode. An example of application where this is useful is provided in section 3.

### 2.4 Algorithm

The minimization of the $\varepsilon$-insensitive loss function (9) can be implemented through the minimization of $n \times N$

positive slack variables $\xi_{ij}$ and $n \times N$ soft constraints of the type

$$-\xi_{ij} - \varepsilon \leq y_i - f_j(\boldsymbol{x}_i) \leq \varepsilon + \xi_{ij}, \qquad (12)$$
$$j = 1, \dots, n, \ i = 1, \dots, N,$$

where $\xi_{ij} \geq 0$. To deal with the switching noise variance assumption, multiple loss functions (11) with different thresholds $\varepsilon$ set to $\delta_j$, $j = 1, \dots, n$, are implemented via the set of constraints

$$-\xi_{ij} - \delta_j \leq y_i - \boldsymbol{w}_j^T \boldsymbol{x}_i - b_j \leq \delta_j + \xi_{ij}, \qquad (13)$$
$$j = 1, \dots, n, \ i = 1, \dots, N,$$

where the submodels $f_j$ have been replaced by (2).

In addition, by following the SVR approach, the submodels $f_j$ are trained by minimizing the norms of the parameter vectors $\|\boldsymbol{w}_j\|_2$. Minimizing these norms, which are also measures of the model complexity, is known to yield models with better generalization performance (Smola and Schölkopf, 2004). The problem of training $n$ models under the decoupling constraints with bounded error (7) is thus approximated by the optimization program involving $n(p+1+N)$ variables:

$$\min_{\boldsymbol{w}_j, b_j, \xi_{ij} \geq 0} \sum_{j=1}^{n} \|\boldsymbol{w}_j\|_2^2 + C \sum_{i=1}^{N} \prod_{j=1}^{n} \xi_{ij} \qquad (14)$$
$$-\xi_{ij} - \delta_j \leq y_i - \boldsymbol{w}_j^T \boldsymbol{x}_i - b_j \leq \delta_j + \xi_{ij},$$
$$j = 1, \dots, n, \ i = 1, \dots, N.$$

Solving this problem with a sufficiently large constant $C$ leads to solution functions $f_j$ in accordance with the original constraints (7). Moreover, the discrete state $\lambda_i$ is readily available for all $i$ from the variables $\xi_{ij}$ vanishing to zero as $\hat{\lambda}_i = j$, for $\xi_{ij} = 0$, i.e. for the model $f_j$ approximating $(\boldsymbol{x}_i, y_i)$ with an error less than $\delta$. The cases where the bounded error constraint is not satisfied, i.e. no $\xi_{ij}$ is zero, can be further discriminated by letting $\hat{\lambda}_i = \arg\min_j(\xi_{ij})$.

In the case of PWA systems where the modes are linearly separable in the regressor space $\mathcal{X}$, undetermined points can be reclassified after the training of separating hyperplanes (the boundaries between the domains $S_j$) based on the determined cases only. The classification issue is not discussed here due to size constraints and the reader is referred to Vapnik (1995) and Mangasarian (2000) for an introduction to state-of-the-art methods.

The hyperparameters of the method are the number $n$ of modes, the bounds $\delta_j$, the regularization parameter $C$ and the number of lagged inputs and outputs (dynamic order). They can be tuned on a subset of the data put aside for validation. When too few data are available, cross-validation techniques can be used. Automatic tuning of the bounds $\delta_j$ is discussed in the following.

### 2.5 Automatic tuning

Using the $\nu$-Support Vector Regression ($\nu$-SVR) formulation described in Schölkopf et al. (2000), one can tune automatically the width $\varepsilon$ of an $\varepsilon$-insensitive loss function. Applying this trick to the algorithm (14) leads to

$$\min_{\boldsymbol{w}_j, b_j, \xi_{ij} \geq 0, \delta_j \geq 0} \sum_{j=1}^{n} \|\boldsymbol{w}_j\|_2^2 + C \sum_{i=1}^{N} \prod_{j=1}^{n} \xi_{ij} + \nu NC \sum_{j=1}^{n} \delta_j^2 \tag{15}$$

$$-\xi_{ij} - \delta_j \leq y_i - \boldsymbol{w}_j^T \boldsymbol{x}_i - b_j \leq \delta_j + \xi_{ij},$$
$$j = 1, \ldots, n, \ i = 1, \ldots, N.$$

Notice that, in comparison to Schölkopf et al. (2000), actually the *squares* of the $\delta_j$ are minimized instead of the $\delta_j$ directly. Minimizing the $\delta_j$ directly would produce a shrinkage effect due a constant influence function equal to one. This is not desirable when dealing with noisy data, since the method relies on the fact that some $\xi_{ij}$, which satisfy $\xi_{ij} \geq \max(0, \ |e_{ij}| - \delta_j)$, are zeros and thus that $e_{ij} \leq \delta_j$ for a certain number of points. On the contrary, in the minimization of a quadratic criterion, the influence function is known to decrease around zero, which leads to non-zero solutions for the $\delta_j$.

As one hyperparameter $\nu$ has been added, the $n$ hyperparameters $\delta_j$ are now automatically tuned. The optimization program (15) thus involves $n(p + 2 + N)$ variables.

This algorithm can be further improved by considering adaptive $\delta_j$ as above but bounded by $\overline{\delta}$, simply by introducing the constraint $\overline{\delta} \geq \delta_j$, for $j = 1, \ldots, n$, in (15). Doing so, we impose a bound $\overline{\delta}$ on the error as in the standard approach (Bemporad et al., 2005), but also allow the algorithm to shrink this bound in order to produce better solutions.

### 2.6 Optimization issues

Problems (14) and (15) are nonlinear optimization programs involving respectively $n(p+1+N)$ and $n(p+2+N)$ variables, which may become a difficulty for a large number of data $N$. However, the proposed method benefits from the good properties of Support Vector Regression and is thus well adapted for the cases where few data are available. Problems minimizing a nonlinear objective function over a convex set, such as (14) and (15), can be solved by general purpose solvers such as the function *fmincon* found in MATLAB, but with no guarantee that the solution is a global minimum. The non-convexity of the objective functions is due to the product terms $\prod_{j=1}^{n} \xi_{ij}$, which have however a global minimum equal to zero (due to $\xi_{ij} \geq 0$). For values of $\delta_j$ so that the bounded error constraints can be satisfied, this optimum is feasible. Local optimization algorithms, such as sequential quadratic programming, are thus expected to converge towards a global minimum and can be used to solve the problems.

In practice, it has been observed that the optimization rarely leads to a local minimum corresponding to a bad solution of the initial system identification problem. Moreover, such bad solutions are easily detected, in which case the optimization can be restarted from another initial point and with a smaller tolerance on the termination criteria. For instance, in the algorithm 1, used in the experiments of section 3.1, bad solutions are detected from the resulting maximum absolute error, $\mathrm{MAE} = \max_{i \in \{1, \ldots, N\}} |y_i - f(x_i)|$. The optimization is restarted with a smaller tolerance Tol, used for the three termination tolerances (change in the variables, change in the objective function and

constraint violation), until the MAE is lower than the maximal error allowed and specified by $\overline{\delta}$. However, when this overall bound is too small compared to the noise level, this condition cannot be satisfied and the algorithm is stopped when Tol reaches a predefined minimal value $m$.

**Algorithm 1.** Computing a solution to problem (15) that is acceptable with respect to the maximal absolute error (MAE).

---

(1) Initialize Tol and its minimal value $m$
(2) Choose a random initial vector of $n \times (p + N + 2)$ variables with values in the interval $[0, 1]$
(3) Call *fmincon* to solve (15)
(4) If MAE $> \overline{\delta}$,
   - if Tol $> m$,
      · Tol $\leftarrow 0.1 \times$ Tol
      · go to step 2
   - else
      · output a warning *"No solution with MAE $< \overline{\delta}$ can be found, maybe the bound $\overline{\delta}$ is too tight for this data set"*

---

## 3. EXAMPLES

Two examples are provided to show the efficiency of the method both for the automatic tuning of the bounds while estimating two synthetic functions and for hybrid system identification.

### 3.1 Switching noise variance and automatic tuning

The first example shows the approximation of a one-dimensional PWA map $y = a_1 x + b_1$, for $x < -2$, and $y = a_2 x + b_2$ otherwise. It also emphasizes the need for different bounds on the errors for the different modes, in case of a switching noise model, and demonstrates the efficiency of the automatic tuning of these bounds.

30 points in the interval $-5 \leq x < -2$ are generated for $a_1 = 1$ and $b_1 = 0.5$ and are corrupted by a uniform noise in the interval $[-\beta_1, \beta_1]$. 41 points in the interval $-2 \leq x \leq 2$ are generated for $a_2 = -0.5$ and $b_2 = 0$ and are corrupted by uniform noise in the interval $[-\beta_2, \beta_2]$. In the first set of experiments, $\beta_1$ and $\beta_2$ are set to 0.8 and 0.2 respectively. All optimization programs are solved by the MATLAB function *fmincon* for $C = 40$. Figure 1 shows the approximation of the $N = 71$ data by algorithm (14) for $\delta_1 = \delta_2 = 0.6$ (top). When using the same bound on the error for the two models, the data are correctly approximated within the bounds, but the bound is clearly too large for the data corrupted by low level noise, leading to $\hat{a}_2 = -0.34$ and $\hat{b}_2 = -0.14$. In this case, using different bounds for each model, in accordance with the noise levels, may lead to a better approximation.

Algorithm (15) is then used with an overall bound $\overline{\delta} = 1$ and $\nu = 1$ to estimate the linear models under the switching noise assumption. As shown at the bottom of Figure 1, this leads to a better fit for $x \geq 2$ with $\hat{a}_2 = -0.49$ and $\hat{b}_2 = -0.01$, while the final bounds on the errors estimated in accordance with the noise levels are $\delta_1 = 0.54$ and $\delta_2 = 0.18$. As a comparison, a manual
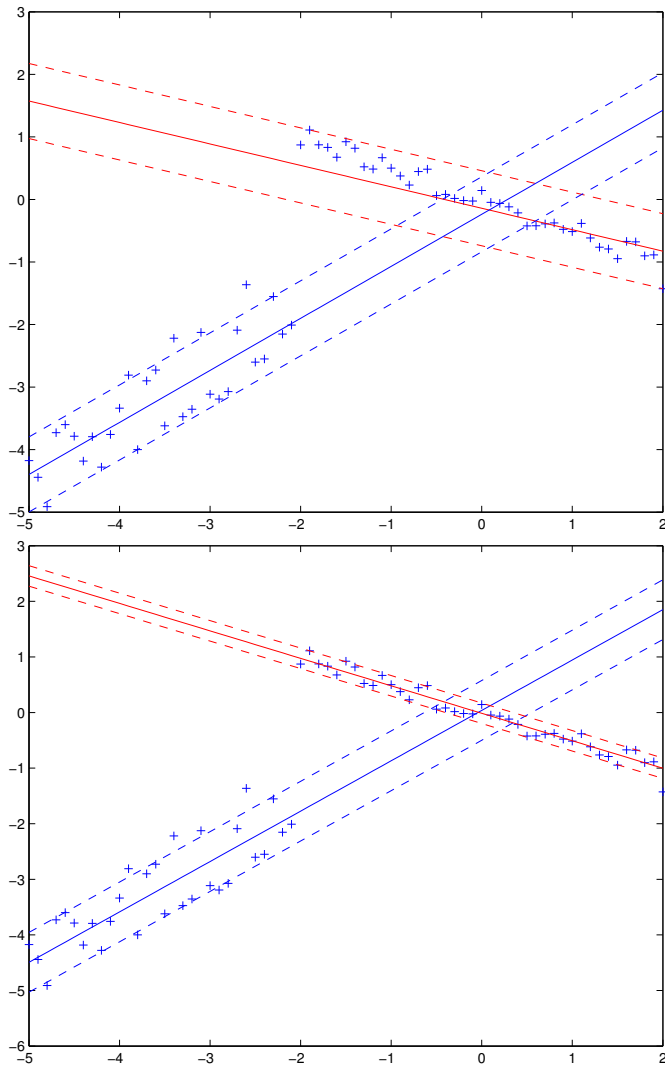
Fig. 1. Simultaneous estimation of 2 linear functions with different noise levels. *Top:* using the same bound on the error for the 2 models, the bound is clearly too large for $x \geq -2$. *Bottom:* automatic tuning of the bounds leads to a better fit.

Table 1. Parameter estimates and Mean Square Error (MSE) for example 1 of section 3.1 and figure 1.

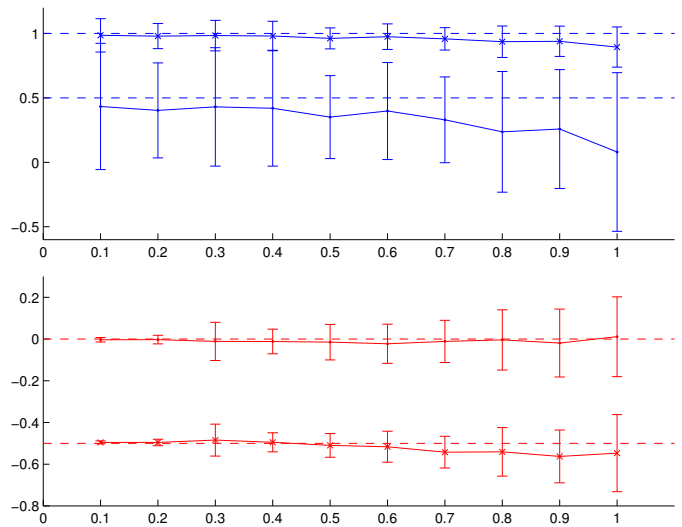|  | $a_1$ | $b_1$ | $a_2$ | $b_2$ | MSE |
|---|---|---|---|---|---|
| *true parameters* | *1* | *0.5* | *-0.5* | *0* | *0.073* |
| same bounds | 0.83 | -0.24 | -0.34 | -0.14 | 0.138 |
| $(\delta_1 = \delta_2 = 0.6)$ |  |  |  |  |  |
| automatic tuning | 0.91 | 0.04 | -0.49 | -0.01 | 0.108 |
| $(\delta_1 = 0.54, \delta_2 = 0.18)$ |  |  |  |  |  |
| manual search | 0.92 | 0.10 | -0.51 | -0.02 | 0.108 |
| $(\delta_1 = 0.6, \delta_2 = 0.1)$ |  |  |  |  |  |



Fig. 2. Average performance of the automatic tuning with respect to the noise level over 100 runs. Expectation of the estimated parameters $a_1$, $b_1$ (top) and $a_2$, $b_2$ (bottom) with their standard deviations versus the noise level $\beta_2$. The true parameters are also represented (dash lines).

grid search of the optimal parameters $\delta_1$ and $\delta_2$ over 10 values in the range $[0.1, 1.0]$ (requiring 100 runs of the algorithm) leads to values close to these: $\delta_1 = 0.6$ and $\delta_2 = 0.2$. Table 1 summarizes the results with respect to the parameter estimates and the Mean Square Error (MSE) given by $MSE = 1/N \sum_{i=1}^{N} (y_i - f(x_i))^2$. The results for the automatic tuning are identical in terms of MSE to the ones obtained by a manual grid search of the optimal values of $\delta_1$ and $\delta_2$. The parameter estimated in these two settings are very close. For the three settings (fixed, manual and automatic tuning), a bias can be observed for the parameter $b_1$, due to a high noise level $\beta_1 = 0.8$ and a low number of data (30 points generated with the parameters $a_1$ and $b_1$).

The automatic tuning is now tested with respect to the variation of the noise level. The noise level for $x < -2$ is set to $\beta_1 = 0.5$, while the one for $x \geq 2$ varies from $\beta_2 = 0.1$ to $\beta_2 = 1$. Figure 2 shows the mean and standard deviations over 100 runs of the the estimated parameters. During the experiments, $\delta_1$ and $\delta_2$ given by the algorithm

(15) evolve in accordance with the noise levels $\beta_1$ and $\beta_2$. Thanks to this automatic tuning, the expectation of the parameter estimates is mostly insensitive to the noise level, while their standard deviations increases only slightly for high levels of noise.

*3.2 Simulated hybrid system identification*

Consider the hybrid system switching between mode 1: $y_t = y_{t-1} + u_{t-1} + e_t$, and mode 2: $y_t = -0.905y_{t-1} + 0.048u_{t-1} + e_t$, where $e_t$ is a zero-mean Gaussian noise of standard deviation 0.1. An output trajectory of 100 points of this system is generated with a random initial condition $y_0$, a random input sequence $u_t \sim \mathcal{N}(0, 1)$ and random mode switches. The MATLAB function *fmincon* is then used to compute the solution of problem (14) for $C = 100$ and $\delta_1 = \delta_2 = 0.1$. The trajectory of the resulting model $\hat{y}_t = f_{\hat{\lambda}_t}(\hat{y}_{t-1}, u_{t-1})$ is shown on Figure 3. The two estimated submodels are $f_1(\hat{y}_{t-1}, u_{t-1}) = 0.989\hat{y}_{t-1} + 1.02u_{t-1}$ and $f_2(\hat{y}_{t-1}, u_{t-1}) = -0.893\hat{y}_{t-1} + 0.043u_{t-1}$, which shows the recovering of the original parameters. The discrete state is estimated by $\hat{\lambda}_t = \arg\min_j(\xi_{tj})$. As shown in Fig. 3, the algorithm recovers the true mode without requiring a settling time and only 5 classification errors occur on the whole trajectory. Moreover, the effect of these classification errors is limited and their origin can
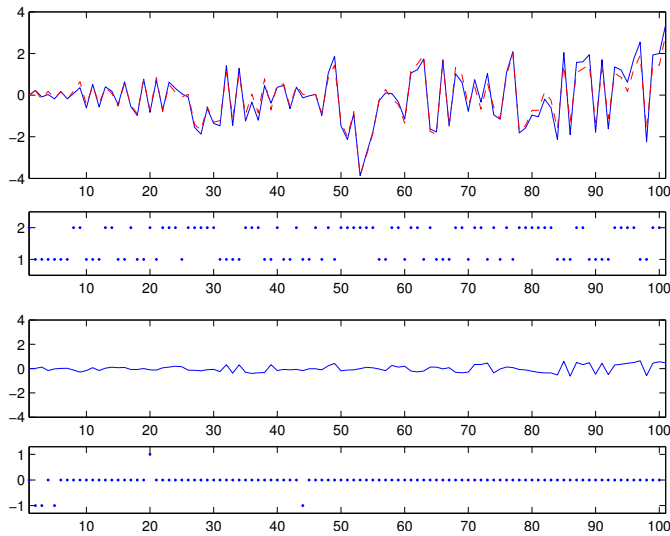
Fig. 3. *From top to bottom*: (a) Trajectory of the system (blue plain line) and of the model (red dash line) in simulation mode (only the initial condition and the input are given to the model); (b) Estimated discrete state $\hat{\lambda}_t$; (c) Output error $y_t - \hat{y}_t$; (d) Classification error $\lambda_t - \hat{\lambda}_t$.

be explained. They occur on ambiguous points for which $f_1(y_{t-1}, u_{t-1}) = f_2(y_{t-1}, u_{t-1}) \pm (\delta_1 + \delta_2)$. In case of a PWA system, these ambiguities could be removed by classifying the points with respect to a separating boundary in the regressor space.

## 4. CONCLUSION

In this paper, a new method for hybrid system identification has been proposed to bridge the gap between the bounded error approach and the algebraic procedure. The method makes use of the $\varepsilon$-insensitive loss function proposed in the machine learning community for Support Vector Regression. Since no assumption on the discrete sequence that generates the data is required, arbitrarily switched ARX (SARX) as well as piecewise ARX (PWARX) models can be identified. Compared to the bounded error approach (Bemporad et al., 2005), the resulting algorithm can deal with switching noise level and automatically tune the error bounds accordingly. It thus offers an alternative, which may be useful when a predefined or preferred model structure is given. Based on support vector regression, the proposed method is also very efficient when few data are available for identification.

Future work will focus on missing topics such as subsytems with different orders, estimation of the number of modes, more experiments including real-life applications, incorporation of prior knowledge and in-depth study of the optimization problems. Besides, extension of the proposed method to nonlinear submodels has been studied by Lauer and Bloch (2008).

## REFERENCES

A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino. A bounded-error approach to piecewise affine system identification. *IEEE Trans. on Automatic Control*, 50 (10):1567–1580, 2005.

P.M.L. Drezet and R.F. Harrison. Support vector machines for system identification. *Proc. of the UKACC Int. Conf. on Control, Swansea, UK*, 1:688–692, 1998.

G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari. A clustering technique for the identification of piecewise affine systems. *Automatica*, 39(2):205–217, 2003.

A. L. Juloski, S. Weiland, and W. Heemels. A Bayesian approach to identification of hybrid systems. *IEEE Trans. on Automatic Control*, 50(10):1520–1533, 2005a.

A.L. Juloski, W. Heemels, G. Ferrari-Trecate, R. Vidal, S. Paoletti, and J. H. G. Niessen. Comparison of four procedures for the identification of hybrid systems. In *Proc. of the Int. Conf. on Hybrid Systems: Computation and Control, Zurich, Switzerland*, volume 3414 of LNCS, pages 354–369. Springer, 2005b.

F. Lauer and G. Bloch. Switched and piecewise nonlinear hybrid system identification. In M. Egerstedt and B. Mishra, editors, *Proc. of the 11th Int. Conf. on Hybrid Systems: Computation and Control (HSCC), St. Louis, MO, USA*, volume 4981 of *LNCS*, pages 330–343. Springer, 2008.

Y. Ma and R. Vidal. Identification of deterministic switched ARX systems via identification of algebraic varieties. In M. Morari and L. Thiele, editors, *Proc. of the 8th Int. Conf. on Hybrid Systems: Computation and Control*, volume 3414 of *LNCS*, pages 449–465, 2005.

O. Mangasarian. Generalized support vector machines. In A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 135–146. MIT Press, Cambridge, MA, USA, 2000.

D. Mattera and S. Haykin. Support vector machines for dynamic reconstruction of a chaotic system. In B. Schölkopf, C. J.C. Burges, and A. J. Smola, editors, *Advances in kernel methods: support vector learning*, pages 211–241. MIT Press, Cambridge, MA, USA, 1999.

J. Roll, A. Bemporad, and L. Ljung. Identification of piecewise affine systems via mixed-integer programming. *Automatica*, 40(1):37–50, 2004.

B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett. New support vector algorithms. *Neural Computation*, 12:1207–1245, 2000.

A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, 2004. ISSN 0960-3174.

V. N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag, New York, NY, USA, 1995.

R. Vidal, S. Soatto, Y. Ma, and S. Sastry. An algebraic geometric approach to the identification of a class of linear hybrid systems. In *Proc. of the 42nd IEEE Conf. on Decision and Control, Maui, Hawaï, USA*, pages 167–172, 2003.