Identification of Hybrid Systems via Mixed-Integer Programming

Alberto Bemporad*, Jacob Roll†, and Lennart Ljung‡

† Department of Electrical Engineering
Linköping University, SE-581 83 Linköping, Sweden
WWW: http://www.control.isy.liu.se
Email: roll, ljung@isy.liu.se

‡ Dip. Ingegneria dell’Informazione
Università di Siena
Via Roma 56, 53100 Siena, Italy
Email: bemporad@dii.unisi.it

* Automatic Control Laboratory
ETH Zürich
ETL I 26, 8092 Zürich, Switzerland
Email: bemporad@aut.ee.ethz.ch

August 24, 2001

Report no.: LiTH-ISY-R-2397
Submitted to CDC’01

Technical reports from the Automatic Control group in Linköping are available by anonymous ftp at the address ftp.control.isy.liu.se. This report is contained in the file 2397.pdf.
Identification of Hybrid Systems via Mixed-Integer Programming
Alberto Bemporad, Jacob Roll, and Lennart Ljung
August 24, 2001

Abstract
This paper addresses the problem of identification of hybrid dynamical systems, by focusing the attention on hinging hyperplanes (HHARX) and wiener piecewise affine (W-PWARX) autoregressive exogenous models. In particular, we provide algorithms based on mixed-integer linear or quadratic programming which are guaranteed to converge to a global optimum. We also discuss issues of state-space realization of HHARX and W-PWARX models into several existing discrete-time hybrid state-space forms.

1 Introduction
Hybrid models describe systems composed by both continuous and discrete components, the former typically associated with physical first principles, the latter with logic devices, such as switches, digital circuitry, software code, etc. A typical instance are real-time systems, where physical processes are controlled by embedded controllers. Their study has recently seen a rapid development, thanks to the interaction between the computer science and the control engineering communities.

Most of the literature of hybrid systems has dealt with the issues of modeling [1,2], and tools have been proposed for stability analysis [3,4], control [2,5,6], verification [7–10], and fault detection [11, 12]. Such tools rely on a model of the hybrid system, which sometimes is either not available because some parameters are unknown, or too complex to be handled. Getting a simple hybrid model from data or from a more complex model is an identification problem, which does not seem to have received enough attention in the hybrid systems community.

On the other hand, other communities dedicated extensive research to develop tools for identification of general nonlinear black-box models (see e.g., [13], where the authors surveyed several nonlinear identification techniques). A few of these techniques lead to piecewise linear (or piecewise affine) models of nonlinear dynamical systems, such as neural networks with piecewise affine perceptrons [14], self-exciting threshold autoregressive (SETAR) models for time-series analysis [15], and hinging hyperplanes models [16,17].

In parallel, the circuit and systems community developed piecewise linear techniques for the representation and the analysis of nonlinear electronic circuits, mostly based on Chua’s canonical piecewise linear representation [18–22].
Because of the equivalence between piecewise affine systems and several classes of hybrid systems [1, 23, 24], such tools for PWA identification can be used to obtain hybrid models, or at least for translating nonlinear/unknown parts of the system into a hybrid form, for which the tools mentioned above can be applied.

As will be pointed out, if the guardlines (i.e., the partition of the PWA mapping) are known, the problem of identifying piecewise affine systems is simple, and can easily be solved using standard techniques. However, when the guardlines are unknown the problem becomes much more difficult. The basic difficulty can be expressed as follows: There are two possibilities: (1) Either a grid defining the cells over which the system is constant is defined a priori or (2) this grid is estimated along with the linear models. The former approach suffers from the curse of dimensionality in the sense that the number of a priori given cells will have to be very large for reasonable flexibility even in the case of moderately many regressors. On the other hand, the estimation of the linear models is easy in this case. The second approach allows for efficient use of fewer cells, but leads to potentially (very) many local minima, which may make it difficult to apply local search (minimization) routines. This dilemma we address in this paper. It should be said right away that we will not offer any practical solution here. Instead we point to reformulations for two subclasses of piecewise affine systems that lead to mixed-integer linear or quadratic programming problems that can be solved for the global optimum. We illuminate this approach with some examples and analysis and also discuss how these insights can be used for new ideas on efficient algorithms.

2 PWARX Models

To begin with, let us consider systems on the form

\[ y(t) = g(\phi(t)) + e(t) \]  

where \( \phi(t) \in \mathbb{R}^n \) is our regression vector, \( e(t) \) is white noise, and \( g \) is a piecewise affine function of the form

\[ g(\phi) = d'_j \phi + c_j \quad \text{if} \quad \bar{H}_j \phi \leq \bar{D}_j \]  

where \( d'_j \in \mathbb{R}^n, c'_j \in \mathbb{R}, \) and the sets \( C_j \triangleq \{ \phi : \ H_j \phi \leq D_j \}, \) \( j = 1, \ldots, s \) are a polyhedral partition of the \( \phi \)-space. To allow for a more compact notation, we let \( \phi(t) = [\phi_1(t) \ldots \phi_n(t)]', \theta_j = [\phi_j c_j]', \) and \( H_j = [-\bar{D}_j \bar{H}_j] \). In this way (2) can be written as

\[ g(\phi) = \varphi' \theta_j \quad \text{if} \quad H_j \varphi \leq 0 \]  

The regression vector could, e.g., consist of old inputs and outputs, i.e., \( \phi(t) = [y(t-1) \ldots y(t-n_a) u(t-1) \ldots u(t-n_b)] \). In this case we call the systems PWARX (PieceWise affine AutoRegressive eXogenous) systems. We do not assume that \( g \) is necessarily continuous over the boundaries, also referred to as guardlines in the hybrid systems literature. Without the hypothesis of continuity over the boundaries, definition (2) is not well posed in general, as the function can be twice (or more times) defined over common boundaries of the sets \( C_j \).
Although one can avoid this issue by replacing some of the “≤” inequalities into “<” in the definition of the regions \( C_j \), this issue is not of practical interest from a numerical point of view.

There are several reasons for focusing on PWARX systems for hybrid systems identification. From a theoretical point of view, piecewise affine systems are equivalent to several classes of linear hybrid systems (see [23] and [1, 24]). From a computational point of view, the advantage in using PWA models is twofold. First, PWA mappings are suitable for efficient identification algorithms. Although this is certainly true also for sigmoidal neural networks and other nonlinear mappings, the second computational advantage comes on the use of the identified models for control, stability analysis, and safety analysis purposes: most of such hybrid tools are only available for PWA or linear hybrid models.

### 2.1 Identification of PWARX Models

Now suppose that we are given \( y(t) \) and \( \varphi(t) \), \( t = 1, \ldots, N \), and want to find the PWARX model that best matches the given data. The identification of model (3) can be carried out by solving the optimization problem

\[
\min \frac{1}{2N} \sum_{k=1}^{N} \left( \sum_{j=1}^{s} \| y_k - \varphi'_k \theta_j \| J_j(\varphi_k) \right) \tag{4a}
\]

subject to

\[
J_j(\varphi_k) = \begin{cases} 
1 & \text{if } H_j \varphi_k \leq 0 \\
0 & \text{otherwise} 
\end{cases} \tag{4b}
\]

+ linear bounds over \( \theta_j, H_j \) \tag{4c}

where \( \theta_j, H_j, j = 1, \ldots, s \) are the unknowns, \( y_k = y(k), \varphi_k = \varphi(k) \) are the estimation data, and \( N \) is the number of data. In (4), we will focus on the 1-norm (\( \| \cdot \| \)) and the squared Euclidean norm (\( \| \cdot \|_2^2 \)), as they allow to express (4) as a mixed-integer linear or quadratic program (MILP/MIQP), respectively, for which efficient solvers exist [25–28]. Note that every norm in (4a) provides a consistent minimum with respect to a probability distribution associated to the noise \( e(t) \), and in particular 1-norm is optimal with respect to exponentially distributed noise (\( p(x) = pe^{-x} \)), 2-norm is optimal with respect to Gaussian noise. We distinguish below among two main cases:

- Case A: \( H_j \) is known, \( \theta_j \) has to be estimated;
- Case B: \( H_j \) is unknown, \( \theta_j \) is unknown.

### 2.2 Known Guardlines

In case A, only \( \theta_j \) have to be determined. This is for instance the case of canonical piecewise linear functions, where the partition of the \( \varphi \)-space is assigned, and only the linear gains \( \theta_j \) in each polyhedral cell have to be determined. If using squared Euclidean norm in (4), we can see that this is an ordinary least-squares problem (or a quadratic program, QP, if there are linear bounds over \( \theta_j \)), and hence can be solved efficiently.
2.3 Unknown Guardlines

Case B is a much harder problem, since it is nonconvex and the objective function generally contains several local minima. However, the optimization problem (4) can be recast as a mixed-integer linear program (MILP) (1-norm or infinity norm) or mixed-integer quadratic program (MIQP) (Euclidean norm).

In the following sections, we focus on two subsets of PWA functions, namely the Hinging Hyperplanes and Wiener processes with PWA static output mapping, and detail the mixed-integer program associated to the identification problem. In general, the complexity of the mixed-integer program needed to solve (4) is related to the number of samples $N$ and regions $s$, and the number of parameters $H_j, D_j, \theta_j$ that are unknown. Note that in general, the guardlines $H_j^T \varphi \leq D_j^T$, (where, given a matrix $M$, $M^i$ denotes the $i$-th row of $M$) cannot be determined exactly from the given estimation data set, as the pairs $y_k, \varphi_k$ are a discrete set of points which, in principle, can be divided by a continuum of possible guardlines.

3 Hinging Hyperplane Models

Hinging hyperplanes (HH) where introduced by Breiman [16] in the context of regression, classification, and function approximation problems. Typically, in nonlinear system identification the family $G$ of models where we look for an optimal approximation is generated by parameterizing a truncated set of basis functions $g_i(\phi), i = 1, \ldots, M$, namely

$$g(\phi) = \sum_{i=1}^{M} \alpha_i g_i(\phi)$$

where $g_i$ have universal approximation properties ($g(\phi) \rightarrow f(\phi)$ as $M \rightarrow \infty$).

In general, $g_i(\phi)$ are obtained by parameterizing a single “mother” function $\kappa(z)$ [13]

$$g_i(\phi) = \kappa(\beta_i(\phi - \gamma_i))$$

and once $M$ has been also fixed, $g^*$ is selected by solving a finite optimization problem over $\alpha_i, \beta_i, \gamma_i, i = 1, \ldots, M$.

One hidden-layer feedforward sigmoidal neural networks corresponds to choosing as mother function the sigmoid $\kappa(z) = \frac{1}{1 + e^{-z}}$. HH models are instead based the hinge function

$$g_i(\phi) = \pm \max\{\beta_i^+ \phi + \gamma_i^+, \beta_i^- \phi + \gamma_i^-\}$$

corresponding to an “open book” in the $(y, \phi)$-space, as depicted in Fig. 1. The $\pm$ sign is needed to represent both convex and nonconvex functions. However, since it will only have a minor effect on the computations in this paper, we will exclude it for the sake of notational simplicity.

By letting $\varphi = [1 \phi]^T$, the HH model can be expressed as

$$g(\varphi) = \sum_{i=1}^{M} \max\{\varphi^T \theta_i^+, \varphi^T \theta_i^-\}$$  \hspace{1cm} (5)
where each pair of hyperplanes “hinges” upon the set \( \{ \varphi : \varphi'(\theta^+ - \theta^-) = 0 \} \). As shown in [17], the original HH model (5) proposed by Breiman [16] is over-parameterized, and can be reduced to

\[
g(\varphi, \Theta) = \varphi'\theta_0 + \sum_{i=1}^{M} \max\{\varphi'\theta_i, 0\}
\]

where

\[
\theta_0 \triangleq \sum_{i=1}^{M} \theta_i^-, \quad \theta_i \triangleq \theta_i^+ - \theta_i^-
\]

and \( \Theta \triangleq [\theta_0 \ldots \theta^M] \in \mathbb{R}^{(n+1) \times M+1} \). Note that (6) corresponds to interpreting (5) as the sum of a linear function and basis functions generated by the mother \( \kappa(z) = \max(z, 0) \). Due to the relation \( -z + \max\{z, 0\} = \max\{-z, 0\} \), \( \forall z \in \mathbb{R} \), there are still redundancies in (6), which can be partially avoided by introducing a requirement on the form

\[
w'\theta_i \geq 0, \quad i = 1, \ldots, M
\]

where \( w \) is any nonzero vector of \( \mathbb{R}^n \), e.g., \( w = \frac{1}{n} \triangleq [1 \ 1 \ldots 1]' \) (or any random vector). (However, vectors \( \theta \) on the hyperplane \( w'\theta = 0 \) can be still mirrored.)

### 3.1 HHARX Models

By setting \( g(\phi) \) in (1) as in (6), we obtain the following HHARX (Hinging-Hyperplane AutoRegressive eXogenous) model

\[
y(t) = \varphi'(t)\theta_0 + \sum_{i=1}^{M} \max\{\varphi'(t)\theta_i, 0\}
\]

\[
\varphi(t) = [1 \ y(t-1) \ldots y(t-n_a) \ u(t-1) \ldots u(t-n_b)]' \in \mathbb{R}^{n+1},
\]
Identiﬁcation Algorithms for HH Models

The ﬁrst algorithm for estimating HH models was proposed by Breiman [16]. Later, in [17] the authors show that the original algorithm is a special case of Newton’s method, and provide a modiﬁcation which guarantees convergence to a local minimum. Other algorithms have been proposed based on tree hinging hyperplane models [29]. In this paper, we propose an alternative approach based on mixed-integer programming, which provides a global minimum, at the price of an increased computational eﬀort with respect to the algorithms in [16,17].

Consider the problem of estimating a HH function of the form (6) from the estimation data set \( \{y_k, \varphi_k\}_{k=1}^N \). We choose the optimal parameters \( \Theta^* \) by solving the mathematical program

\[
\Theta^* \triangleq \arg \min_{\Theta} V(\Theta) \triangleq \sum_{k=1}^N |y_k - g(\varphi_k, \Theta)| \quad (9a)
\]

subject to

\[
\begin{align*}
\theta^j - \theta_j & \leq \theta^j \quad & j = 1, \ldots, M \\
\sum_{i} \theta_i & \geq 0, \quad i = 1, \ldots, M \quad (9b)
\end{align*}
\]

where the inequalities in (9b) are componentwise and optionally included for reﬂecting partial a priori information about the range of the parameters \( \theta_i \). As mentioned above, the reason for using the absolute value in (9) is mainly numerical, because as we will show below the problem can be recast as a mixed-integer linear program (MILP) (see also [30]). Another reasonable possibility is to use the squared Euclidean norm \(|y_k - g(\varphi_k, \Theta)|^2\), as the problem can be recast as a mixed integer quadratic program (MIQP). The problem can be also recast as an MILP by using inﬁnity norm over time (i.e. \( \max_{k=1, \ldots, N} \) instead of \( \sum_{k=1}^N \)), although this would be highly sensitive to possible outliers in the estimation data, as mentioned above.

4.1 Optimization Problem

4.1.1 MILP Formulation

The optimization problem (9) can be recast as an MILP by introducing additional integer 0-1 and continuous variables. To this end, we associate one 0-1 variable \( \delta_{ik} \) to the sign of \( \varphi_k^i \theta_i \)

\[
[\delta_{ik} = 0] \Rightarrow [\varphi_k^i \theta_i \leq 0], \quad i = 1, \ldots, M, \quad k = 1, \ldots, N \quad (10)
\]

and introduce new continuous variables \( z_{ik} \)

\[
z_{ik} = \max\{\varphi_k^i \theta_i, 0\} = \varphi_k^i \theta_i \delta_{ik} \quad (11)
\]

The relations (10) and (11) can be transformed into mixed-integer linear inequalities, by using standard techniques [6,31–33]. By assuming that the bounds over \( \theta_i \) are all ﬁnite, Eq. (10) is equivalent to

\[
\begin{align*}
\varphi_k^i \theta_i & \leq M_{ik}^\theta \delta_{ik} \quad (12a) \\
\varphi_k^i \theta_i & \geq \varepsilon + (m_{ik}^\theta - \varepsilon)(1 - \delta_{ik}) \quad (12b)
\end{align*}
\]
where $\varepsilon$ is a small positive scalar (e.g., the machine precision),

$$
M^\circ_{ik} \triangleq \|\varphi_k\|_\infty M^\circ_i \left( \geq \max_{\theta_i^- \leq \theta_i \leq \theta_i^+} \varphi_k^i \theta_i \right)
$$

$$
m^\circ_{ik} \triangleq -M_{ik} \left( \leq \min_{\theta_i^- \leq \theta_i \leq \theta_i^+} \varphi_k^i \theta_i \right)
$$

and

$$
M^\circ_i \triangleq \max_{\theta_i^- \leq \theta_i \leq \theta_i^+} \varphi_k^i \theta_i
$$

Similarly, (11) is equivalently rewritten as

$$
-M^\circ_{ik} \delta_{ik} + z_{ik} \leq 0 \quad (13a)
$$

$$
m^\circ_{ik} \delta_{ik} - z_{ik} \leq 0 \quad (13b)
$$

$$
m^\circ_{ik}(1 - \delta_{ik}) + z_{ik} \leq \varphi_k^i \theta_i \quad (13c)
$$

$$
-M^\circ_{ik}(1 - \delta_{ik}) - z_{ik} \leq -\varphi_k^i \theta_i \quad (13d)
$$

Finally, by introducing auxiliary slack variables $\epsilon_k \geq |y_k - g(\varphi_k, \Theta)|$, $k = 1, \ldots, N$, (9) we can prove the following

**Proposition 1** The optimum of problem (9) is equivalent to the optimum of the following MILP

$$
\begin{align*}
\min \quad & \sum_{k=1}^{N} \epsilon_k \\
\text{subj. to} \quad & \epsilon_k \geq y_k - \left( \varphi_k^i \theta_0 + \sum_{i=1}^{N} z_{ik} \right) \quad (14b) \\
\epsilon_k & \geq \varphi_k^i \theta_0 + \sum_{i=1}^{N} z_{ik} - y_k \quad (14c) \\
(12), \ (13), \ (7) \quad (14d)
\end{align*}
$$

**Proof.** Clearly, given the data $\{y_k, \varphi_k\}$ and for any $\Theta$, there exist only one set of variables $z_{ik}, \delta_{ik}$ satisfying constraints (12), (13), and for any $\{\epsilon_k\}$ compatible with the constraints (14b)-(14c) it results $\sum_{i=1}^{N} \epsilon_k \geq V(\Theta) \geq V(\Theta^*)$, where $\Theta^*$ is the solution to (9). In particular, for the optimizer $\epsilon^\#_1, \Theta^\#$ of (14) it results $\sum_{i=1}^{N} \epsilon^\#_k \geq V(\Theta^\#) \geq V(\Theta^*)$. On the other hand, the values $\epsilon^*_k \triangleq |y_k - g(\varphi_k, \Theta^*)|$, $\delta^*_0 \triangleq 0$ if $\varphi_k^i \theta < 0$, or 1 otherwise, and $z^*_{ik} \triangleq \delta_{ik} \varphi_k^i \theta_i, i = 1, \ldots, M$, $k = 1, \ldots, N$, are admissible for problem (14), which proves that the minimum of (14) $V(\Theta^\#) = \sum_{i=1}^{N} \epsilon^\#_k \leq \sum_{i=1}^{N} \epsilon^*_k \leq V(\Theta^*)$. Therefore, $V(\Theta^\#) = V(\Theta^*)$. $\square$
Example 1  Consider the following HHARX model
\[ y(t) = 0.8y(t-1) + 0.4u(t-1) - 0.1 + \]
\[ + \max\{ -0.3y(t-1) + 0.6u(t-1) + 0.3, 0 \} \] (15)

The model is identified on the data reported in Fig. 3(a), by solving an MILP
with 66 variables (of which 20 integers) and 168 constraints. The problem is
solved by using Cplex 6.5 [28] (1014 LP solved in 0.68 s on a Sun Ultra 10
running Matlab 5.3), and, for comparison, is solved again using BARON [26]
(73 LP solved in 3.00 s, same machine), which results in a zero output prediction
error (Fig. 3(b)). The fitted HH model is reported in Fig. 2. By adding a random
Gaussian noise with zero mean and variance 0.1 on the measured output
\[ y(t) \]

the following model
\[ y(t) = 0.83y(t-1) + 0.34u(t-1) - 0.20 + \]
\[ + \max\{ -0.34y(t-1) + 0.62u(t-1) + 0.40, 0 \} \] (16)
is identified in 1.39 s (3873 LP solved) using Cplex (7.86 s, 284 LP using
BARON) on the estimation set reported in Fig. 4(a), and produces the validation
data reported in Fig. 4(b). For comparison, we identified the linear ARX model
\[ y(t) = 0.82y(t-1) + 0.72u(t) \] (17)
on the same estimation data, obtaining the validation data reported in Fig. 5
(higher order ARX models did not produce significant improvements). Note
that the output error on the validation data is computed by feeding the ARX
model with the outputs obtained by the PWARX model (15). Clearly, the error
generated by driving the ARX model in open-loop with the validation input
\[ u(t) \]

is much larger, and would not make (17) suitable for instance for formal
verification tools, where a good performance of open-loop prediction is a critical
requirement.

4.1.2 MIQP Formulation

When the squared 2-norm is used in the objective function,
\[ V(\Theta) \triangleq \sum_{k=1}^{N} (y_k - g(\varphi_k, \Theta))^2 \] (18)
the optimization problem can be recast as the MIQP
\[ \min_{\theta_0, \ldots, \theta_M, \delta_{11}, \ldots, \delta_{MN}, z_{11}, \ldots, z_{MN}} \sum_{k=1}^{N} (y_k - (\varphi_k^T \theta_0 + \sum_{i=1}^{M} z_{ik}))^2 \] (19a)
subj. to (12), (13), (7) (19b)

Note that the problem is not strictly positive definite, for instance the cost
function does not depend on \( \theta_i, \delta_{ik} \) (which only appear in the constraints). For
numerical reasons, the Hessian associated to the MIQP (19) is augmented with
the term \( \sigma I \), where \( \sigma \) is a small fixed number.
Estimation data and identified HH model

Figure 2: Identification of model (15) – noiseless case. Identified HH model

(a) Estimation data

(b) Validation data

Figure 3: Identification of model (15) – noiseless case

4.1.3 Complexity

Despite the good solvers available, the complexity of the MILP or MIQP problems is well known to be \( NP \)-hard, and in particular it is exponential in the number \( MN \) of binary variables. Therefore, the mixed-integer approach is computationally affordable only for model with a few data, or if data are clustered together (e.g., 100 data are averaged into 10 data). Otherwise, local search methods [16,17] are preferable, although they do not guarantee the best fit (i.e., a global minimum to the optimization problem).

Example 2 Consider again the PWARX system (15). In Figure 6 we compare the performance in terms of LP/QPs and total computation time of the linear criterion (14) versus the quadratic criterion (19). The reported numbers are computed on a Sun Ultra 60 (2 x 360 MHz) running Matlab 5.3 and the solver BARON [26], by averaging the result of ten estimation data sets generated by feeding random Gaussian inputs \( u(t) \) and zero output noise to system (15).
4.2 Discontinuous HHARX Models

In HHARX models, the output \( y(t) \) is a continuous function of the regressor \( \phi(t) \). On the other hand, hybrid systems often consist of piecewise affine discontinuous mappings. In order to tackle discontinuities, we can modify the HH model (6) in the form

\[
g(\varphi, \Theta) = \varphi' \theta_0 + \sum_{i=1}^{M} (\varphi' \theta_i + a_i) \delta_i(\varphi) \tag{20a}
\]

\[
[\delta_i(\varphi) = 0] \leftrightarrow [\varphi' \theta_i \leq 0], \ i = 1, \ldots, M, \ k = 1, \ldots, N \tag{20b}
\]

where \( a_i, i = 1, \ldots, M \) are additional free parameters, \( a_i^- \leq a_i \leq a_i^+ \), or, more
in general, in the form

$$g(\varphi, \Theta) = \varphi' \theta_0 + \sum_{i=1}^{M} (\varphi' \theta_i) \delta_i(\varphi)$$

$$[\delta_i(\varphi) = 0] \leftrightarrow [\varphi' \mu_i \leq 0], \ i = 1, \ldots, M, \ k = 1, \ldots, N$$

(21a)

(21b)

where $\mu_i$, $i = 1, \ldots, M$ are additional free vectors of parameters, $\mu_i^- \leq \mu_i \leq \mu_i^+$, $1^T \mu_i \geq 0$. Similarly to (14), by introducing new continuous variables $z_{ik}$ both the identification problems (20) and (21) can be again recast as an MILP. With respect to (14), the MILP has $\mu_i$ or $a_{ik}$, $i = 1, \ldots, M$ as additional optimization variables. Note that the problem in general does not have a unique solution. For instance, once $\theta_i$, $a_{ik}$, $z_{ik}$, $\delta_{ik}$ have been fixed, there exist in general infinitely many vectors $\mu_i$ satisfying the constraints (21b).

### 4.3 Robust HHARX Models

In formal verification methods, model uncertainty needs to be handled in order to provide a safety guarantees. Typically, the model is associated with a bounded uncertainty. In the context of symbolic solvers for timed automata, differential inclusions $a \leq \dot{x} \leq b$ are handled, for instance by the solver Hytech [34]. For piecewise affine and MLD systems, the verification algorithm proposed in [10] handles model uncertainty as additive input disturbances entering a nominal model.

In the present context of HHARX models, we wish to find an uncertainty description of the form

$$g(\varphi(t), \Theta^-) \leq y(t) \leq g(\varphi(t), \Theta^+), \ \forall t \geq 0$$

(22)
for an inclusion-type of description, or the form
\[ g(t) = g(\varphi(t), \Theta^*) + n(t), \ n^- \leq n(t) \leq n^+ \] (23)
for an additive-disturbance-type of description. Clearly, since the model is identified from a finite estimation data set, fulfillment of (22) or (23) for all \( t \) and for all initial conditions cannot be guaranteed, unless additional hypotheses on the model which generates the data are assumed.

Nevertheless, a pair of extreme models \( \Theta^-, \Theta^+ \) can be obtained by solving (14) or (19) with the additional linear constraints
\[ y_k \geq g(\varphi_k, \Theta), \ \forall k = 1, \ldots, N \] (24)
for estimating \( \Theta^- \), and
\[ y_k \leq g(\varphi_k, \Theta), \ \forall k = 1, \ldots, N \] (25)
for estimating \( \Theta^+ \), respectively. An additive-disturbance description can be instead computed in two alternative ways:

1. First, identify a model \( \Theta^* \) by solving (14) or (19) and then compute
\[ n^+ \triangleq \max_{k=1,\ldots,N} y_k - g(\varphi_k, \Theta^*) \]
\[ n^- \triangleq \min_{k=1,\ldots,N} y_k - g(\varphi_k, \Theta^*) \] (26)

2. Modify the MILP (14) by setting replacing \( \epsilon_k \) with one variable \( \epsilon \) only, and minimize \( \epsilon \). The corresponding optimum \( \epsilon^* \) provides a nominal model such that the bound on the norm of the additive disturbance \( n(t) \) is minimized.

## 5 Piecewise Affine Wiener Models

Let us now turn to the class of models shown in Fig. 7, described by the relations
\[ A(z)x_t = B(z)u_t \]
\[ y_t = f(x_t) \] (27a)
where \( A(z) = 1 + \sum_{i=1}^{n_a} a_i z^{-i} \), \( B(z) = \sum_{i=1}^{n_b} b_i z^{-i} \), and \( z^{-1} x_t = x_{t-1} \). We assume that \( f(x) \) is a piecewise affine, invertible function (without restrictions we can assume that \( f \) is strictly increasing), and parameterize its inverse as
\[ x_t = y_t - \alpha_0 + \sum_{i=1}^{M} \pm \max \{ \beta_i y_t - \alpha_i, 0 \} \] (27b)
Both signs ± are allowed in order to be able to represent nonconvex functions. We assume that the number \( M^+ \) of positive signs is known (without restrictions we can let these be the first terms of the sum). As \( \max\{-z, 0\} = -z + \max\{z, 0\} \) for all \( z \in \mathbb{R} \), without loss of generality we can also assume \( \beta_i \geq 0 \).

6 Identification of W-PWARX Models

As seen from Fig. 7, a Wiener model consists of a linear dynamic system followed by an output nonlinearity. In some cases, the two can be identified separately: first the inverse nonlinearity is estimated by supplying a quasi-static input, and then a linear dynamic model is identified by using standard linear techniques [35]. On the other hand, in some other cases the input signal cannot be designed arbitrarily, as input/output estimation data are simply supplied by other sources. Then one algorithm which estimates the whole Wiener process is desirable. Here, we describe an algorithm based on mixed-integer programming, which identifies W-PWARX models of the form (27). Such PWA form is particularly useful when the identified system models an unknown part of a larger hybrid model. We assume that we are given an estimation data set \( \{y_t, u_t\}_{t=1}^N \).

Define \( a_h = a_h^+ - a_h^- \), \( a_h^+, a_h^- \geq \gamma \), where \( \gamma > 0 \) is any positive scalar. Then

\[
a_h \max\{\beta_i y_{t-h} - \alpha_i, 0\} = \\
= \max\{a_h^+ \beta_i y_{t-h} - a_h^- \alpha_i, 0\} - \\
- \max\{a_h^- \beta_i y_{t-h} - a_h^+ \alpha_i, 0\} \\
= \max\{c_{ih}^+ y_{t-h} - d_{ih}^+, 0\} - \max\{c_{ih}^- y_{t-h} - d_{ih}^-, 0\}
\]

where

\[
c_{ih}^\pm \triangleq a_h^\pm \beta_i, \\
d_{ih}^\pm \triangleq a_h^\pm \alpha_i, \quad i \in [1, M], \ h \in [1, n_a]
\]

Let also

\[
c_{i0} = c_{i0}^+ = c_{i0}^- \triangleq \beta_i \\
d_{i0} = d_{i0}^+ = d_{i0}^- \triangleq \alpha_i \\
d_{ih} \triangleq a_h \alpha_0 \\
d_{i0} \triangleq \alpha_0 \\
d_{0h} \triangleq \sum_{h=0}^{n_a} d_{ih} = \left( 1 + \sum_{h=1}^{n_a} a_h \right) \alpha_0
\]

For each \( - \max \) and \( + \max \) function in (27b), and for each \( t \), we introduce the integer variables \( \delta_{it} \in \{0, 1\} \)

\[
[\delta_{it} = 1] \iff [\beta_i y_{t-h} - \alpha_i \geq 0], \ \ i \in [1, M], \ t \in [1, N]
\]

Without loss of generality, we can assume that the \( M^+ \) first breakpoints in the PWA output nonlinearity are ordered, and similarly for the \( M - M^+ \) last breakpoints. Clearly, the logic constraint

\[
[\delta_{it} = 1] \quad \rightarrow \quad [\delta_{jt} = 1]
\]
should hold for all \( i, j \leq M^+ \) such that \( j < i \), and for all \( i, j > M^+ \) such that \( j < i \). Each constraint (29) is translated into

\[
\delta_{it} - \delta_{jt} \leq 0,
\]

(30)

and a minimal set of inequalities is obtained by collecting (30) only for pairs of consecutive indices \( i, j \). Moreover, since the output data \( y_t \) can be ordered, we can also get additional relations on \( \delta_{it} \) by using (28). In fact, if \( \delta_{it_0} = 1 \) and \( y_{t_1} > y_{t_0} \), it must follow that \( \delta_{it_1} = 1 \). We can translate these relations into

\[
\delta_{it_0} - \delta_{it_1} \leq 0, \quad \forall t_1 \neq t_0 : y_{t_1} \geq y_{t_0}
\]

(31)

As \( a^+_{ih}, a^-_{ih} > 0 \), from (28) it also follows

\[
[\delta_{it} = 1] \leftrightarrow [c^+_{ih} y_t - d^+_{ih} \geq 0]
\]

(32)

Let us also introduce the auxiliary continuous variables

\[
z_{it0} = (c_{i0} y_t - d_{i0}) \delta_{it}
\]

\[
z_{ih} = [(c^+_{ih} - c^-_{ih}) y_{ih} - (d^+_{ih} - d^-_{ih}) \delta_{i(t-h)}], \quad h \in [1, n_a]
\]

(33)

Using the same techniques as in (12) and (13), we can translate (32) and (33) to linear inequalities.

Now,

\[
x_t = y_t - d_{00} + \sum_{i=1}^{M} \pm z_{it0}
\]

(34)

\[
a_h x_t = a_h y_{t-h} - d_{0h} + \sum_{i=1}^{M} \pm z_{ih}
\]

By (27) and (34),

\[
x_t = y_t - d_{00} + \sum_{i=1}^{M} \pm z_{it0} = \sum_{k=1}^{n_h} b_k u_{t-k} - \\
- \sum_{h=1}^{n_a} \left( a_h y_{t-h} - d_{0h} + \sum_{i=1}^{M} \pm z_{ih} \right)
\]

which provides the relation

\[
y_t = - \sum_{h=1}^{n_a} a_h y_{t-h} + \sum_{k=1}^{n_h} b_k u_{t-k} + d_{0} - \sum_{i=1}^{M} \sum_{h=0}^{n_a} \pm z_{ith}
\]

(35)

In order to fit the estimation data to model (35), we solve the mixed-integer quadratic program (MIQP)

\[
\begin{align*}
\min & \frac{1}{N} \sum_{t=1}^{N} \left| y_t + \sum_{h=1}^{n_a} a_h y_{t-h} - \\
- \sum_{k=1}^{n_h} b_k u_{t-k} - d_{0} + \sum_{i=1}^{M} \sum_{h=0}^{n_a} \pm z_{ith} \right|^2 \\
\text{subj. to linear constr. from (30), (31), (32), and (33)}
\end{align*}
\]

(36)
with respect to the variables $a_h$, $b_k$, $c_{i0}$, $d_{i0}$, $\tilde{d}_0$, $\tilde{c}_{ih}$, $d_{ih}^\perp$, $z_{ih}$, and the binary variables $\delta_{ih}$. The solution to (36) provides the optimal parameters $a_h^*$, $b_k^*$, and $\alpha_{i0}^* \equiv \frac{d_{i0}^* - c_{i0}^*}{1 + \sum_{n=1}^{I_h} \alpha_n^*}$, $\alpha_{i}^* \equiv d_{i0}^*$, $\beta_{i}^* \equiv c_{i0}^*$. Finally, we can obtain the estimation $f^*(x)$ by inverting (27b) (see [36] for details).

Example 2. A Wiener model constituted by a first-order linear system and a nonlinearity with two breakpoints is identified, using $N = 20$ estimation data points. The system is first identified using noiseless data, and then using noisy measurements $\bar{y}_t = y_t + \epsilon_t$, where $\epsilon_t$ are independent and uniformly distributed on a symmetric interval around 0. The MIQP problem (36) is solved by running BARON [26] on a Sun Ultra 10. The resulting estimates are shown in Table 1. The estimated parameters are overall very close to the true values, the closer the

| Par. | True | $\epsilon_t = 0$ | $|\epsilon_t| < 0.01$ | $|\epsilon_t| < 0.1$ |
|------|------|-----------------|-----------------|-----------------|
| $a_1$ | -0.5 | -0.5000         | 0.5000          | 0.4990          | 0.4990          |
| $b_1$ | 2    | 2.0000          | 2.0024          | 2.0003          |
| $a_0$ | -2   | -2.0000         | -2.0000         | -1.7748         |
| $a_1$ | 0.5  | 0.5000          | 0.5095          | 0.5016          | 0.5028          |
| $a_2$ | -1.5 | -1.5000         | -1.4924         | -1.4999         |
| $\beta_1$ | 0.5 | 0.5000          | 0.5016          | 0.5028          |
| $\beta_2$ | 0.5 | 0.5000          | 0.4988          | 0.4876          |
| CPU   | -    | 45.44 s         | 51.33 s         | 90.34 s         |

Table 1: Estimation results

lower the intensity of the output noise, as it should be expected. The estimated model was also tested on a set of validation data, and we report in Fig. 8 the resulting one-step-ahead predicted output and output error. Note that such a good performance cannot be achieved by using standard linear identification techniques.
6.1 Complexity Analysis

By imposing the constraints expressed by (30) and (31), the degrees of freedom for the integer variables, and hence the complexity, are reduced considerably. In fact, instead of having to test $2^{MN}$ different cases in the worst case, only $\binom{M+N}{M} \cdot \binom{M}{M}$ combinations would be tested. For example, for $N = 20$ and $M = 2$ this means that the number of possible combinations of integer variables decreases from approximately $10^{12}$ to 462. In general, for a fixed $M$ the worst-case complexity grows as $N^M$. Note that this simplification is possible since the nonlinearity is one-dimensional, which allows an ordering of the breakpoints and of the output data.

7 State-Space Realizations

Similarly to the linear ARX case, assuming that the identified polynomials $A(z)$, $B(z)$ are coprime and $n_a \geq n_b$, a minimal state-space realization containing $n_a + n_b$ can be obtained for HHARX models, by using different hybrid state-space discrete-time paradigms introduced recently in the hybrid systems literature. For W-PWARX, a minimal state-space realization containing $n_a$ states can be obtained.

7.1 MLD Realization

Mixed Logical Dynamical (MLD) systems [6] are a discrete-time hybrid formalism which allows specifying the evolution of continuous variables through linear dynamic equations, of discrete variables through propositional logic statements and automata, and the mutual interaction between the two. The key idea of the approach consists of embedding the logic part in the state equations by transforming Boolean variables into 0-1 integers, and by expressing the relations as mixed-integer linear inequalities, similarly to what was done in (12), (13), and (30). Applications that can be naturally modeled within the MLD framework are reported in [6,37,38].

The MLD model has the form

\begin{align}
\xi(t+1) &= \Phi \xi(t) + G_1 u(t) + G_2 \delta(t) + G_3 z(t) \\
y(t) &= H \xi(t) + D_1 u(t) + D_2 \delta(t) + D_3 z(t) \\
E_2 \delta(t) + E_3 z(t) &\leq E_4 u(t) + E_5 \xi(t) + E_5
\end{align}

where $\xi \in \mathbb{R}^{n_c} \times \{0,1\}^{n_r}$ is a vector of continuous and binary states, $u \in \mathbb{R}^{m_c} \times \{0,1\}^{m_r}$ are the inputs, $y \in \mathbb{R}^{p_c} \times \{0,1\}^{p_r}$ the outputs, $\delta \in \{0,1\}^{r_c}$, $z \in \mathbb{R}^{r_c}$ represent auxiliary binary and continuous variables respectively, which are introduced when transforming logic relations into mixed-integer linear inequalities, and $\Phi$, $G_1$, $G_2$, $G_3$, $H$, $D_1$, $D_2$, $D_3$, $E_4$, $E_5$ are matrices of suitable dimensions.

**Proposition 2** HHARX models (8) admit an MLD state-space realization with $n_a + n_b$ states.

**Proof.** The MLD state-space realization of (8) can be immediately obtained by defining auxiliary binary variables $\delta_i(t)$, $z_i(t)$ similarly to what was done
in (12) (13), and setting \( \xi(t) = [y(t - 1) \ldots y(t - n_a) \ u(t - 1) \ldots u(t - n_b)]' \in \mathbb{R}^{n_a+n_b} \).

Proposition 3 W-PWARX models (27) admit an MLD state-space realization with \( n_a \) states.

Proof. Let

\[
\begin{align*}
\xi(t) &= \Phi \xi(t-1) + G_1 u(t) \\
x(t) &= C \xi(t)
\end{align*}
\]

(38)

be a canonical state-space realization of (27a), and consider the output nonlinearity

\[
y(t) = C \xi(t) - \bar{\alpha}_0 + \sum_{i=1}^{M} \pm \max\{\tilde{\beta}_i C \xi(t) - \bar{\alpha}_i, 0\}.
\]

(39)

Define \( M \) auxiliary binary variables

\[
[\delta_i(t) = 1] \iff [\tilde{\beta}_i C \xi(t) - \bar{\alpha}_i \geq 0]
\]

and \( M \) continuous \( z_i(t) = (\tilde{\beta}_i C \xi(t) - \bar{\alpha}_i) \delta_i(t) \). By using translations into mixed-integer inequalities similar to (12) and (13), the MLD matrices \( H, D_2, D_3, E_2, \ldots, E_5 \) can be immediately obtained.

7.2 PWA Realization

Analogously to what defined in (2), a PWA state-space system is defined as

\[
\begin{align*}
\xi(t+1) &= A_j \xi(t) + B_j u(t) + f_j, \quad \text{for } \xi(t) \in C_j \triangleq \{\xi : H_j \xi \leq D_j\}
\end{align*}
\]

(40)

where \( \xi \in \mathbb{R}^n \), and \( \{C_j\}_{j=0}^{s-1} \) is a polyhedral partition of the state-space (more in general, the partition is defined in the augmented state+input space and a direct feedthrough of the input on the output can be present, although this is not the case for the HHARX and W-PWARX models considered in this paper).

The following Propositions can be obtained as a corollary of the equivalence between MLD and PWA systems [24], and allows to construct a PWA state-space realization of (27) via the above MLD realization.

Proposition 4 HHARX models (8) admit a PWA state-space realization (40) with \( n_a + n_b \) states and at most \( 2^M \) regions.

Proposition 5 W-PWARX models (27) admit a PWA state-space realization (40) with \( n_a \) states and at most \( 2^M \) regions.

7.3 MMPS Realization

Max-Min-Plus-Scaling (MMPS) state space models where introduced in [1] and have the form

\[
\begin{align*}
\xi(t+1) &= M_{\xi}(\xi(t), u(t), d(t)) \\
y(t) &= M_{y}(\xi(t), u(t), d(t)) \\
c &\geq M_{c}(\xi(t), u(t), d(t))
\end{align*}
\]

(41)
where $\mathcal{M}_x$, $\mathcal{M}_y$, $\mathcal{M}_c$ are MMPS expressions (i.e., expressions defined by the composition of max and min functions, sum, and multiplication by a scalar) in terms of the state $\xi(t)$, the input $u(t)$, and the auxiliary variables $d(t)$, which are all real-valued. When $\mathcal{M}_c$ is empty, we refer to as unconstrained MMPS systems.

**Proposition 6** HHARX models (8) admit a MMPS state-space realization (41) with $n_a + n_b$ states.

**Proof.** As for the MLD realization, define
\[
\xi(t) = \begin{bmatrix} y(t-1) & \ldots & y(t-n_a) & u(t-1) & \ldots & u(t-n_b) \end{bmatrix}^T \in \mathbb{R}^{n_a + n_b}. Then,
\]
\[
\xi(t+1) = \begin{bmatrix}
I_{n_a-1} & 0 \\
0 & I_{n_b-1}
\end{bmatrix}
\begin{bmatrix}
\mathcal{M}_y(\xi(t), u(t)) \\
\xi(t) + e_{n_a+1} u(t)
\end{bmatrix} + e_{n_a+1} u(t)
\]
\[
y(t) = \mathcal{M}_y(\xi(t), u(t))
\]
where $\mathcal{M}_y$ is directly obtained by (8) ($\mathcal{M}_c$, $d(t)$ are empty), and $e_{n_a+1} = [0 \ldots 1 0 \ldots 0]^T$ is the $(n_a + 1)$-th column of the identity matrix $I_{n_a+n_b}$.

**Proposition 7** W-PWARX models (27) admit an unconstrained MMPS state-space realization (41) with $n_a$ states.

**Proof.** Simply follows by combining the linear state-space realization (38) and the MMPS expression (39). \qed

### 7.4 LC and ELC Realization

Linear Complementary (LC) and Extended Linear Complementary (ELC) state-space realizations can be obtained by exploiting the equivalences described in [1].

### 8 Conclusions

In this paper we have addressed the problem of identification of hybrid dynamical systems, by focusing our attention on piecewise affine (PWARX), hinging hyperplanes (HHARX), and Wiener piecewise affine (W-PWARX) autoregressive exogenous models. In particular, for the two latter classes we have provided globally convergent algorithms based on mixed-integer linear or quadratic programming. Several problems remain open, such as the choice of persistently exciting input signals $u$ for identification (i.e., that allow for the identification of all the affine dynamics), and criteria like Akaike’s criterion for choosing the best order and number of hinging pairs in HHARX models. Future research will also be devoted to faster suboptimal algorithms, by combining MIP solvers and clustering techniques. We have also discussed issues of state-space realization of HHARX models into several existing discrete-time hybrid state-space models (MLD, PWA, MMPS, LC, ELC).

The hybrid systems community has recently dedicated substantial attention to develop tools for modeling, stability analysis, control synthesis, verification, and fault detection, which strongly rely on the availability of a hybrid model of the process at hand. We hope that this paper attracts attention on the problem of estimating such a hybrid model from data.
References


