Identification of Hybrid Systems via Mixed-Integer Programming

Alberto Bemporad[†§] Jacob Roll[*] Lennart Ljung[*]

[†] *Automatic Control Laboratory*
ETH Zurich
ETL I 26, 8092 Zurich, Switzerland
tel: +41 1 6326679, fax: +41-1-6321211
`bemporad@aut.ee.ethz.ch`

[*] *Division of Automatic Control*
Linköping University
S-581 83 Linköping, Sweden
tel: +46 13 281310, fax: +46 13 282622
`roll,ljung@isy.liu.se`

[§] *Dip. Ingegneria dell'Informazione*
Università di Siena
Via Roma 56, 53100 Siena, Italy
tel: +39 0577 234631, fax : +39 0577 234632
`bemporad@dii.unisi.it`

Tech. Report AUT00-28

October 24, 2000

Abstract

Many successful tools have been proposed by several researchers for safety analysis, stability analysis, and control/scheduling synthesis for hybrid systems. Such tools rely on a hybrid model of the process under investigation. Whenever the hybrid model is not available because some parameters are unknown, or too complex to be handled, getting a simple hybrid model from data is an identification problem, which has not yet received much attention in the hybrid systems community. In this paper we focus on the identification of piecewise affine (PWA) systems. PWA systems are a well studied class of hybrid systems, that was also shown to be equivalent to arbitrary interconnections of logic and linear dynamic components. We present in detail an algorithm for identifying piecewise affine Wiener models from input-output data. Two notable properties of the algorithm are that the globally optimal solution is found, and that the complexity is polynomial with respect to the number of data.

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 [1–3], and tools have been proposed for stability analysis [4,5], control [3,6,7], verification [8–11], and fault detection [12,13]. 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., [14], where the authors surveyed several nonlinear identification techniques). Although most of these techniques are not immediately applicable in a hybrid context because the resulting model is highly nonlinear, a few techniques lead to piecewise linear (or piecewise affine) models of nonlinear dynamical systems, such as neural networks with piecewise affine perceptrons [15], self-exciting threshold autoregressive (SETAR) models for time-series analysis [16], and hinging hyperplanes models [17, 18].

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 [19–23].

Because of the equivalence between piecewise affine systems and several classes of hybrid systems [1, 24, 25], 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.

In this paper, we introduce the class of PWARX (Piecewise-affine AutoRegressive eXogenous) systems for system identification, and show that the the problem is simple if the *guardlines* (i.e., the polyhedral partition of the PWA mapping) are known (this is the case for instance of the canonical piecewise linear representation [22]), while it is much more complex if also the guardlines have to be determined. In order to attack the problem algorithmically, we focus our attention on W-PWARX (Wiener Piecewise-affine AutoRegressive eXogenous) systems as hybrid linear discrete-time models, by showing how mixed-integer quadratic programming can be used to guarantee a global minimal identification error. We believe that the method is particularly useful for identifying unknown nonlinear dynamical parts of a larger process, when the goal is to construct a linear hybrid model of the whole process itself.

2 PWARX Models

Autoregressive models rely on the assumption that $y(t)$ only depends on a certain number of past inputs and outputs $\varphi(t) = [y(t-1)\ \ldots\ y(t-n_a)\ u(t-1)\ \ldots\ u(t-n_b)]' \in \mathbb{R}^n$, $n \triangleq n_a + n_b$, where n_a, n_b are parameters decided by the user (criteria exist for choosing n_a, n_b, like Akaike's criterion [26], false nearest neighbors [27], etc.). The vector $\varphi(t)$ is denoted as regression vector, and for simplicity of notation, we have assumed that the system is SISO. In nonlinear system black-box identification, the aim is at determining a function $f(\varphi)$ such that $y(t) = f(\varphi(t))$ well fits a given estimation data set $\{u(t), y(t)\}_{t=0}^{T-1}$, e.g., $\sum_{t=0}^{T-1} \|y(t) - f(\varphi(t))\|$ is minimized (if f is partially known, for instance because of some physical insight, we refer to this as *graybox* identification). In order to render a finite dimensional (and therefore computationally tractable) optimization problem, we restrict our attention to a family of functions $\mathcal{G}$, parameterize such a family by a vector of real valued parameters θ, and look for the best θ^* whose corresponding function $g^* \in \mathcal{G}$ is the best approximation of f in $\mathcal{G}$ according to the selected fitting criterion.

In this paper we consider the family $\mathcal{G}$ of piecewise affine functions

$$g(\varphi) = d^j\varphi + c^j \quad \text{if} \quad H_j\varphi \leq D_j \tag{1}$$

where $d^j \in \mathbb{R}^n$, $c^j \in \mathbb{R}$, and the sets $\mathcal{C}_j \triangleq \{\varphi :\ H_j\varphi \leq D_j\}$, $j = 1, \ldots, s$ are a polyhedral partition of the φ-space. 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 (1) is not well posed in general, as the function can be twice (or more times) defined over common boundaries of the sets $\mathcal{C}_j$. Although one can avoid this issue by replacing some of the "$\leq$" inequalities into "$<$" in the definition of the regions $\mathcal{C}_j$, this issue is not of practical interest from a numerical point of view.

We define the PWARX (Piecewise-affine AutoRegressive eXogenous) model as

$$y(t) = g(\varphi(t)) + e(t) \tag{2}$$

where g is a piecewise affine (not necessarily continuous) function of the form (1), and $e(t)$ is an additive term accounting for the fact that $y(t)$ is not exactly a function of past inputs and data. When g is uniquely defined over the whole φ-space, the trajectories of the PWARX system (2) are well defined for all t.

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, as shown in [24] and [1, 25], where constructive proofs are provided. 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

Model (2) can be more explicitly rewritten in the form

$$y(t) = \sum_{k=0}^{n_a} a_k^j y(t-k) + \sum_{k=1}^{n_b} b_k^j u(t-k) + c^j + e(t) \quad \text{if} \quad H_j\varphi(t) \leq D_j \tag{3}$$

where $j = 1, \dots, s$, and let $\theta_j = [a_1^j \ \dots \ a_{n_a}^j \ b_1^j \ \dots \ b_{n_b}^j \ c^j]' \in \mathbb{R}^{n+1}$.

The identification of model (3) can be carried out by solving the optimization problem

$$\min \frac{1}{2N} \sum_{k=1}^{N} \left(\sum_{j=1}^{s} \|y_k - \varphi_k' \theta_j\| \mathcal{J}_j(\varphi_k) \right) \tag{4a}$$

$$\text{subj. to } \mathcal{J}_j(\varphi_k) = \begin{cases} 1 & \text{if } H_j\varphi_k \leq D_j \\ 0 & \text{otherwise} \end{cases} \tag{4b}$$

$$+ \text{ linear bounds over the components of } \theta_j,\ H_j,\ D_j \tag{4c}$$

where θ_j, H_j, D_j $j = 1, \dots, s$ are the unknowns, $y_k = y(k-1)$, $\varphi_k = \varphi(k-1)$ are the estimation data, and $N = T$ is the number of data. In (4), we will focus on the squared Euclidean norm ($\|\cdot\|_2^2$), as it allows to express (4) as a mixed-integer quadratic program (MIQP), for which efficient solvers exist [28–30]. It can be also mentioned that if the 1-norm or the infinity norm is used in (4), this can be cast as a mixed-integer linear program (MILP) [31]. 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 ($\rho(x) = \rho e^{-x}$), 2-norm is optimal with respect to Gaussian noise. We distinguish below among two main cases:

- Case A: H_j, D_j are known, θ_j has to be estimated;
- Case B: H_j and/or D_j are unknown, θ_j is unknown.

2.2 Known Guardlines

In (4), only θ_j have to be determined. This is for instance the case of canonical piecewise linear functions, where the partition of the φ-space is assigned, and only the linear gains θ_j in each polyhedral cell have to be determined.

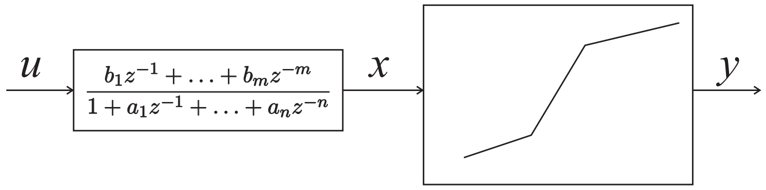


Figure 1: Wiener process with piecewise affine static output mapping

Given H_j, D_j, $j = 1, \ldots, s$, the identification algorithm is based on the following optimization problem

$$\Theta^* \triangleq \arg\min \frac{1}{2N} \sum_{k=1}^{N} \left(\sum_{j=1}^{s} \|y_k - \varphi_k' \theta_j\| \mathcal{J}_j(\varphi_k) \right) \tag{5a}$$

$$\text{subj. to } \theta_j^- \le \theta_j \le \theta_j^+ \tag{5b}$$

where $\Theta \triangleq [\theta_1 \ \ldots \ \theta_s] \in \mathbb{R}^{(n+1)\times s}$ are the unknowns, and $\mathcal{J}_j(\varphi_k)$ are numbers computed directly from the estimation data.

Let $I^j = \{k \le N : \ H_j \varphi_k \le D_j\}$ the set of time instants the regression vector φ_k lies in region $\#j$ (i.e., the indicator function $\mathcal{J}_j(\varphi_k) = 1$ iff $k \in I^j$). Then the cost function in (5a) can be rewritten as

$$\frac{1}{2N}\Theta' \begin{bmatrix} \sum_{k\in I^1} \varphi_k \varphi_k' & & \\ & \ddots & \\ & & \sum_{k\in I^q} \varphi_k \varphi_k' \end{bmatrix} \Theta +$$
$$-\frac{1}{N} \left[\ \sum_{k\in I^1} y_k' \varphi_k \quad \cdots \quad \sum_{k\in I^q} y_k' \varphi_k \ \right] \Theta + \frac{1}{2N} \sum_{k=1}^{N} y_k' y_k \ .$$

The problem has clearly a block diagonal structure (both the cost function and the constraints over Θ), and hence the problem can be separated by identifying each subsystem j independently via linear system identification techniques [26].

2.3 Unknown Guardlines

The optimization problem (4) can be recast as a mixed-integer linear program (MILP) (infinity norm) or mixed-integer quadratic program (MIQP) (Euclidean norm). In the following section, we focus on a subset of PWARX systems, namely 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 T and regions s, and the number of parameters H_j, D_j, θ_j that are unknown. Note that in general, the guardlines $H_j^i \varphi \le D_j^i$, (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, φ_k are a discrete set of points which, in principle, can be divided by a continuum of possible guardlines.

3 Piecewise Affine Wiener Models

We consider the class of models shown in Fig. 1, described by the relations

$$\begin{array}{rcl} A(z)x(t) & = & B(z)u(t) \\ y(t) & = & f(x(t)) \end{array} \tag{6a}$$

where $A(z) = 1 + \sum_{l=1}^{n_a} a_l z^{-l}$, $B(z) = \sum_{l=1}^{n_b} b_l z^{-l}$, and z^{-1} is the delay operator, $z^{-1}x(t) = x(t-1)$. 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

$$x(t) = y(t) - \alpha_0 + \sum_{i=1}^{M} \pm \max\{\beta_i y(t) - \alpha_i, 0\} \ . \tag{6b}$$

Both signs $+$ and $-$ are allowed in order to be able to represent nonconvex functions. As $\max(-z, 0) = -z + \max(z, 0)$ for all $z \in \mathbb{R}$, without loss of generality we can also assume $\beta_i \geq 0$, .

4 Identification of W-PWARX Models

As seen from Fig. 1, 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 [26]. 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. In such cases, the identification cannot be split, and one algorithm which estimates the whole Wiener process is desirable. Here below, we describe an algorithm based on mixed-integer programming, which identifies W-PWARX models of the form (6). Such PWA form is particularly useful when the identified system models an unknown part of a larger hybrid model.

Given an estimation data set $[y(t), u(t)]$, $t = 0, \ldots, N-1$, the identification task is to find the set of parameters b_i, a_i, α_i, β_i which best fits such data.

Define $a_h = a_h^+ - a_h^-$, $a_h^+, a_h^- \geq \gamma$, where $\gamma > 0$ is any positive scalar. Here we let $\gamma = 1$. Then

$$\begin{aligned}
&a_h \max\{\beta_i y(t-h) - \alpha_i, 0\} = \\
&\quad = (a_h^+ - a_h^-) \max\{\beta_i y(t-h) - \alpha_i, 0\} = \\
&\quad = \max\{a_h^+ \beta_i y(t-h) - a_h^+ \alpha_i, 0\} - \max\{a_h^- \beta_i y(t-h) - a_h^- \alpha_i, 0\} \\
&\quad = \max\{c_{ih}^+ y(t-h) - d_{ih}^+, 0\} - \max\{c_{ih}^- y(t-h) - d_{ih}^-, 0\}
\end{aligned}$$

where

$$\begin{aligned}
c_{ih}^{\pm} &\triangleq a_h^{\pm} \beta_i, \ i \in [1, M], \ h \in [1, n_a] \\
d_{ih}^{\pm} &\triangleq a_h^{\pm} \alpha_i, \ i \in [1, M], \ h \in [1, n_a]
\end{aligned}$$

Let also

$$\begin{aligned}
c_{i0} &= c_{i0}^+ = c_{i0}^- \triangleq \beta_i, \ i \in [1, M] \\
d_{i0} &= d_{i0}^+ = d_{i0}^- \triangleq \alpha_i, \ i \in [1, M] \\
d_{0h} &\triangleq a_h \alpha_0, \ h \in [1, n_a] \\
d_{00} &\triangleq \alpha_0,
\end{aligned}$$

and

$$\bar{d}_0 \triangleq \sum_{h=0}^{n_a} d_{0h} = \left(1 + \sum_{h=1}^{n_a} a_h\right) \alpha_0 \ .$$

For each $-\max$ and $+\max$ function in (6b), and for each time instant t, we introduce the integer variables $\delta_{it} \in \{0, 1\}$

$$[\delta_{it} = 1] \ \leftrightarrow \ [\beta_i y(t) - \alpha_i \geq 0], \ i \in [1, M], \ t \in [0, N-1]. \tag{7}$$

Let $\mathcal{I}^+$, $\mathcal{I}^-$ be the sets of indices $i \in [1, M]$ corresponding to positive max functions and negative max functions, respectively. Without loss of generality, we can assume that the breakpoints $i \in \mathcal{I}^+$ in the PWA output nonlinearity are ordered, and similarly for $i \in \mathcal{I}^-$. Clearly, the logic constraint

$$[\delta_{it} = 1] \quad \rightarrow \quad [\delta_{jt} = 1] \tag{8}$$

should hold for all $i, j \in \mathcal{I}^+$ such that $j < i$, and for all $i, j \in \mathcal{I}^-$ such that $j < i$. Each constraint (8) is translated into

$$\delta_{it} - \delta_{jt} \leq 0, \tag{9}$$

and a minimal set of inequalities is obtained by collecting (9) only for pairs of indices i, j consecutive in $\mathcal{I}^+$, and i, j consecutive in $\mathcal{I}^-$. Moreover, since the output data $y(t)$ can be easily ordered, we can also get additional relations on δ_{it} by using (7). In fact, if $\delta_{it_0} = 1$ and $y(t_1) > y(t_0)$, it must follow that $\delta_{it_1} = 1$. Like above, we can translate these relations into

$$\delta_{it_0} - \delta_{it_1} \leq 0, \quad \forall t_1 \neq t_0 : y(t_1) \geq y(t_0), \quad t_0 \in [1, N] \; . \tag{10}$$

As $a_h^+, a_h^- > 0$, from (7) it also follows

$$[\delta_{it} = 1] \;\; \leftrightarrow \;\; \left[a_h^{\pm}\beta_i y(t) - a_h^{\pm}\alpha_i \geq 0\right], \; i \in [1, M], \; h \in [0, n_a], \; t \in [0, N-1]$$

and hence

$$[\delta_{it} = 1] \;\; \leftrightarrow \;\; \left[c_{ih}^{\pm} y(t) - d_{ih}^{\pm} \geq 0\right], \; i \in [1, M], \; h \in [0, n_a], \; t \in [0, N-1]. \tag{11}$$

The above conditions can be translated as linear inequalities [7]

$$\begin{array}{rcl} c_{ih}^{\pm} y(t) - d_{ih}^{\pm} & \geq & \bar{M}_{ith}^{+}(\delta_{it} - 1) \\ c_{ih}^{\pm} y(t) - d_{ih}^{\pm} & \leq & -\varepsilon + (\varepsilon - \bar{m}_{ith})\delta_{it} \end{array} \tag{12}$$

where $\bar{M}_{ith}$, $\bar{m}_{ith}$ are upper- and lower-bounds on $c_{ih}^{\pm} y(t) - d_{ih}^{\pm}$, respectively, which can be computed from the output data and a priori bounds on α_i, β_i, a_h, and ε is a small positive number (e.g., the machine precision). Introduce the auxiliary continuous variables

$$z_{ith}^{\pm} \triangleq (c_{ih}^{\pm} y(t-h) - d_{ih}^{\pm})\delta_{i(t-h)}, \; i \in [1, M], \; h \in [0, n_a], \; t \in [\max\{n_a, n_b\}, N-1] \tag{13}$$

with $z_{it0}^{+} = z_{it0}^{-} = z_{it0}$ where (13) can be transformed into linear inequalities of the form [7]

$$\begin{array}{rcl} z_{ith}^{\pm} & \leq & \bar{M}_{i(t-h)h}\delta_{it} \\ -z_{ith}^{\pm} & \leq & -\bar{m}_{i(t-h)h}\delta_{it} \\ z_{ith}^{\pm} & \leq & c_{ih}^{\pm} y(t-h) - d_{ih}^{\pm} - \bar{m}_{i(t-h)h}(1 - \delta_{it}) \\ -z_{ith}^{\pm} & \leq & -c_{ih}^{\pm} y(t-h) + d_{ih}^{\pm} + \bar{M}_{i(t-h)h}(1 - \delta_{it}) \; . \end{array} \tag{14}$$

Now,

$$\begin{array}{rcl} x(t) & = & y(t) - d_{00} + \displaystyle\sum_{i=1}^{M} \pm z_{it0} \\ a_1 x(t-1) & = & a_1 y(t-1) - d_{01} + \displaystyle\sum_{i=1}^{M} \pm (z_{it1}^{+} - z_{it1}^{-}) \\ \vdots & & \vdots \\ a_{n_a} x(t - n_a) & = & a_{n_a} y(t - n_a) - d_{0n_a} + \displaystyle\sum_{i=1}^{M} \pm (z_{itn_a}^{+} - z_{itn_a}^{-}) \; . \end{array} \tag{15}$$

By (6) and (15),

$$\begin{array}{rcl} x(t) & = & y(t) - d_{00} + \displaystyle\sum_{i=1}^{M} \pm z_{it0} \\ & = & -\displaystyle\sum_{h=1}^{n_a} \left(a_h y(t-h) - d_{0h} + \displaystyle\sum_{i=1}^{M} \pm (z_{ith}^{+} - z_{ith}^{-}) \right) + \displaystyle\sum_{k=1}^{n_b} b_k u(t-k) \end{array}$$

which provides the relation

$$\begin{aligned} y(t) = &-\sum_{h=1}^{n_a} a_h y(t-h) + \sum_{k=1}^{n_b} b_k u(t-k) + \bar{d}_0 \\ &-\sum_{i=1}^{M} \left(\pm z_{it0} + \sum_{h=1}^{n_a} \pm (z_{ith}^+ - z_{ith}^-) \right) \ . \end{aligned} \tag{16}$$

In order to fit the estimation data to model (16), we solve the mixed-integer quadratic program (MIQP)

$$\min \frac{1}{N} \sum_{t=\max\{n_a,n_b\}}^{N-1} \Bigg| y(t) + \sum_{h=1}^{n_a} a_h y(t-h) - \sum_{k=1}^{n_b} b_k u(t-k) - \bar{d}_0 \tag{17a}$$

$$+ \sum_{i=1}^{M} \left(\pm z_{it0} + \sum_{h=1}^{n_a} \pm (z_{ith}^+ - z_{ith}^-) \right) \Bigg|^2$$

$$\text{subj. to constraints (9), (12), (14)} \tag{17b}$$

with respect to the continuous variables $\{a_h, b_k, c_{i0}, d_{i0}, \bar{d}_0, c_{ih}^{\pm}, d_{ih}^{\pm}, z_{ith}^{\pm}\}$ and the binary variables $\{\delta_{it}\}$, $h \in [1, n_a]$, $i \in [0, M]$, $k \in [1, n_b]$, $t \in [0, N-1]$. The solution to (17) provides the optimal parameters a_h^*, b_h^*, and

$$\begin{aligned} \alpha_0^* &\triangleq \frac{\bar{d}_0^*}{1 + \sum_{h=1}^{n_a} a_h^*} \\ \alpha_i^* &\triangleq d_{i0}^* \\ \beta_i^* &\triangleq c_{i0}^* \ . \end{aligned}$$

Finally, since the nonlinearity $f(x)$ is invertible, we can obtain the estimation $f^*(x)$ by using the result of the following Lemma:

Lemma 1 *Consider the function*[1]

$$\begin{aligned} f(x) = x - \alpha_0 + \sum_{i=1}^{M} s_i \max\{\beta_i x - \alpha_i, 0\} \ , \\ \beta_i > 0, \quad \frac{\alpha_1}{\beta_1} < \frac{\alpha_2}{\beta_2} < \cdots < \frac{\alpha_M}{\beta_M} \ , \end{aligned} \tag{18}$$

where $s_i \in \{-1, 1\}$, *and assume that* f *is strictly increasing, i.e.,*

$$0 < \sum_{m=0}^{k} s_m \beta_m < \infty \quad \forall k = 1, \ldots, M \ . \tag{19}$$

Then the inverse of $f(x)$ *is*

$$\begin{aligned} f^{-1}(y) &= y + \alpha_0 - \sum_{k=1}^{M} s_k \max\{\tilde{\beta}_i y - \tilde{\alpha}_i, 0\} \\ \tilde{\beta}_i &= \frac{\beta_k}{\sum_{m=0}^{k-1} s_m \beta_m \sum_{m=0}^{k} s_m \beta_m} \\ \tilde{\alpha}_i &= \frac{\sum_{m=0}^{k-1} s_m (\beta_m \alpha_k - \beta_k \alpha_m)}{\sum_{m=0}^{k-1} s_m \beta_m \sum_{m=0}^{k} s_m \beta_m} \ . \end{aligned} \tag{20}$$

Proof. See Appendix A. □

[1]For the sake of a more compact notation, we let $s_0 = \beta_0 = 1$.

4.1 An Example

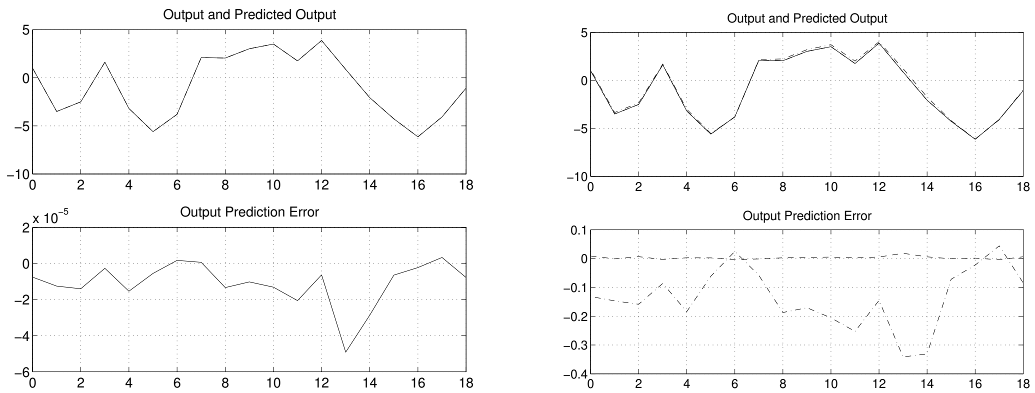
A Wiener model constituted by a first-order linear system cascaded by a nonlinearity with two breakpoints is identified, using $N = 20$ estimation data points. The system is first identified using noiseless data, and then using noisy measurements $\tilde{y}(t) = y(t) + e(t)$, where $e(t)$ are independent random variables, uniformly distributed on a symmetric interval around 0. The MIQP problem (17) is solved by running BARON [29] on a Sun Ultra 10. The resulting estimated parameters are shown in Table 1.

Table 1: Estimation results

Parameter	True value	Noiseless data	$\lvert e(t)\rvert < 0.01$	$\lvert e(t)\rvert < 0.1$
a_1	-0.5	-0.5000	-0.4990	-0.5360
b_1	2	2.0000	2.0024	2.0003
α_0	-2	-2.0000	-2.0001	-1.7748
α_1	0.5	0.5000	0.5095	0.5509
α_2	-1.5	-1.5000	-1.4924	-1.4999
β_1	0.5	0.5000	0.5016	0.5028
β_2	0.5	0.5000	0.4988	0.4876
CPU time	-	45.44 s	51.33 s	90.34 s

The estimated parameters are overall very close to the true values, the closer the 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. 2 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.



(a) System estimated with noiseless data.

(b) System estimated with output noise $|e(t)| \leq 0.01$ (dashed line), and $|e(t)| \leq 0.1$ (dot-dashed line)

Figure 2: Validation results

4.2 Complexity Analysis

The complexity of the problem is tightly connected to the number of integer variables, which equals MN. However, by imposing the constraints expressed by (9) and (10), the degrees of freedom for the integer variables are reduced considerably. In fact, instead of having to test 2^{MN} different cases in the *worst case*, only $\binom{M+N}{M}$ 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 231. 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.

5 State-Space Realizations of W-PWARX Models

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 states can be obtained for W-PWARX models, by using different hybrid state-space discrete-time paradigms introduced recently in the hybrid systems literature.

5.1 MLD Realization

Mixed Logical Dynamical (MLD) systems [7] 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 (9), (12), and (14). Applications that can be naturally modeled within the MLD framework are reported in [7, 32, 33].

The MLD model has the form

$$\xi(t+1) = \Phi\xi(t) + \mathcal{G}_1 u(t) + \mathcal{G}_2\delta(t) + \mathcal{G}_3 z(t) \tag{21a}$$

$$y(t) = \mathcal{H}\xi(t) + \mathcal{D}_1 u(t) + \mathcal{D}_2\delta(t) + \mathcal{D}_3 z(t) \tag{21b}$$

$$\mathcal{E}_2\delta(t) + \mathcal{E}_3 z(t) \leq \mathcal{E}_1 u(t) + \mathcal{E}_4\xi(t) + \mathcal{E}_5 \tag{21c}$$

where $\xi \in \mathbb{R}^{n_c} \times \{0,1\}^{n_\ell}$ is a vector of continuous and binary states, $u \in \mathbb{R}^{m_c} \times \{0,1\}^{m_\ell}$ are the inputs, $y \in \mathbb{R}^{p_c} \times \{0,1\}^{p_\ell}$ the outputs, $\delta \in \{0,1\}^{r_\ell}$, $z \in \mathbb{R}^{r_c}$ represent auxiliary binary and continuous variables respectively, which are introduced when transforming logic relations into mixed-integer linear inequalities, and Φ, $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{G}_3$, $\mathcal{H}$, $\mathcal{D}_1,\ldots,\mathcal{D}_3$, $\mathcal{E}_1,\ldots,\mathcal{E}_5$ are matrices of suitable dimensions.

Proposition 1 *W-PWARX models (6) admit an MLD state-space realization with n_a states.*

Proof. See Appendix B. □

5.2 PWA Realization

Analogously to what defined in (1), a PWA state-space system is defined as

$$\begin{array}{rcl} \xi(t+1) & = & A_j\xi(t) + B_j u(t) + f_j \\ y(t) & = & C_j\xi(t) + g_j \end{array}, \text{ for } \xi(t) \in \mathcal{C}_j \triangleq \{\xi : \ H_j\xi \leq D_j\} \tag{22}$$

where $\xi \in \mathbb{R}^n$, and $\{\mathcal{C}_j\}_{j=0}^{s-1}$ 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 W-PWARX models considered in this paper).

The following Proposition 2 can be obtained as a corollary of the equivalence between MLD and PWA systems [25], and allows to construct a PWA state-space realization of (6) via the above MLD realization.

Proposition 2 *W-PWARX models (6) admit a PWA state-space realization (22) with n_a states and at most 2^M regions.*

Remark 1 hile Wiener PWA processes can be represented as PWA systems, the converse is clearly false.

5.3 MMPS Realization

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

$$\begin{array}{rcl} \xi(t+1) & = & \mathcal{M}_\xi(\xi(t), u(t), d(t)) \\ y(t) & = & \mathcal{M}_y((\xi(t), u(t), d(t)) \\ c & \geq & \mathcal{M}_c(\xi(t), u(t), d(t)) \end{array} \tag{23}$$

where $\mathcal{M}_\xi$, $\mathcal{M}_y$, $\mathcal{M}_c$ 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 3 *W-PWARX models (6) admit an unconstrained MMPS state-space realization (23) with n_a states.*

Proof. Simply follows by combining the linear state-space realization (30) and the MMPS expression (31). □

Remark 2 e can also rewrite (6) as

$$\begin{aligned} y(t) = f(x(t)) = \\ &= f((1-A(z))x(t) + B(z)u(t)) = \\ &= f\left((1-A(z))f^{-1}(y(t)) + B(z)u(t)\right) \end{aligned} \tag{24}$$

By using the expression of $f(x)$ that we get from Lemma 1, we get $y(t)$ explicitly as a function of $y(t-h)$, $u(t-h)$. Clearly, this is a two-layer input/output unconstrained MMPS model (i.e., two levels of nesting of max functions).

5.4 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].

6 Conclusions

We have presented an algorithm for identifying piecewise affine Wiener models from input-output data. Two notable properties of the algorithm are that the globally optimal solution is found, and that for a given the number of breakpoints in the nonlinearity the complexity is polynomial (instead of exponential) with respect to the number of data. Future research will be devoted to the extension of the algorithm to the MIMO case (although it is not yet clear if the polynomial property can be still guaranteed), and to faster suboptimal algorithms, by combining MIP solvers and clustering techniques.

References

[1] W.P.M.H. Heemels, B. De Schutter, and A. Bemporad. Equivalences among classes of hybrid dynamical models. Technical report, October 2000. submitted HS'01.

[2] G. Labinaz, M.M. Bayoumi, and K. Rudie. A survey of modeling and control of hybrid systems. *Annual Reviews of Control*, 21:79–92, 1997.

[3] M.S. Branicky, V.S. Borkar, and S.K. Mitter. A unified framework for hybrid control: model and optimal control theory. *IEEE Trans. Automatic Control*, 43(1):31–45, 1998.

[4] M. Johansson and A. Rantzer. Computation of piece-wise quadratic Lyapunov functions for hybrid systems. *IEEE Trans. Automatic Control*, 43(4):555–559, 1998.

[5] M.S. Branicky. Multiple Lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Trans. Automatic Control*, 43(4):475–482, April 1998.

[6] J. Lygeros, C. Tomlin, and S. Sastry. Controllers for reachability specifications for hybrid systems. *Automatica*, 35(3):349–370, 1999.

[7] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, March 1999.

[8] R. Alur, T.A. Henzinger, and P.-H. Ho. Automatic symbolic verification of embedded systems. *IEEE Trans. on Software Engineering*, 22:181–201, 1996.

[9] A. Chutinan. *Hybrid System Verification Using Discrete Model Approximations.* PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University, May 1999.

[10] S. Kowalewski, O. Stursberg, M. Fritz, H. Graf, I. Hoffmann, J. Preußig, M. Remelhe, S. Simon, and H. Treseler. A case study in tool-aided analysis of discretely controlled continuos systems: the two tanks problem. In *Hybrid Systems V*, volume 1567 of *Lecture Notes in Computer Science*, pages 163–185. Springer-Verlag, 1999.

[11] A. Bemporad, F.D. Torrisi, and M. Morari. Optimization-based verification and stability characterization of piecewise affine and hybrid systems. In B. Krogh and N. Lynch, editors, *Hybrid Systems: Computation and Control*, volume 1790 of *Lecture Notes in Computer Science*, pages 45–58. Springer Verlag, 2000.

[12] J. Lunze. Diagnosis of quantized systems based on a timed discrete-event model. *IEEE Trans. on Systems, Man & Cybernetics, Part A*, 30(3):322–335, May 2000.

[13] A. Bemporad, D. Mignone, and M. Morari. Moving horizon estimation for hybrid systems and fault detection. In *Proc. American Contr. Conf.*, 1999.

[14] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.Y. Glorennec, H. Hjalmarsson, and A. Juditsky. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724, 1995.

[15] C.A. Lehalle and R. Azencott. Piecewise affine neural networks and nonlinear control. In *Proc. 8th Int. Conf. on Artificial Neural Networks (ICANN 98)*, volume 2, pages 633–638, 1998.

[16] M.C. Medeiros, A. Veiga, and M.G.C. Resende. A combinatorial approach to piecewise linear time series analysis. Technical Report 99.5.1, AT&T Labs Research, Florham Park, NJ 07932 USA, 1999.

[17] L. Breiman. Hinging hyperplanes for regression, classification, and function approximation. *IEEE Transactions on Information Theory*, 39(3):999–1013, May 1993.

[18] P. Pucar and J. Sjöberg. On the hinge-finding algorithm for hinging hyperplanes. *IEEE Trans. Inform. Theory*, 44(3):1310–1319, May 1998.

[19] L.O. Chua and A. Deng. Canonical piecewise-linear representation. *IEEE Trans. Circuits Syst. I*, 35:511–525, 1988.

[20] C. Kahlert and L.O. Chua. The complete canonical piecewise-linear representation — part i: The geometry of the domain space. *IEEE Trans. Circuits Syst. I*, 39:222–236, 1992.

[21] P. Julián, A. Desages, and O. Agamennoni. High level canonical piecewise linear representation using a simplicial partition. *IEEE Trans. Circuits Syst. I*, 46:463–480, 1999.

[22] P. Julián. *A High Level Canonical Piecewise Linear Representation: Theory and Applications.* Phd thesis, Universidad National del Sur, Bahía Blanca, 1999.

[23] P. Julian, M. Jordan, and A. Desages. Canonical piecewise-linear approximation of smooth functions. *IEEE Trans. Circuits and Systems — I: Fundamental Theory and Applications*, 45(5):567–571, May 1998.

[24] E.D. Sontag. Interconnected automata and linear systems: A theoretical framework in discrete-time. In R. Alur, T.A. Henzinger, and E.D. Sontag, editors, *Hybrid Systems III - Verification and Control*, number 1066 in Lecture Notes in Computer Science, pages 436–448. Springer-Verlag, 1996.

[25] A. Bemporad, G. Ferrari-Trecate, and M. Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE Trans. Automatic Control*, to appear. `http://control.ethz.ch/`.

[26] L. Ljung. *System Identification : Theory for the User.* Prentice Hall, 2^{nd} edition, 1999.

[27] M. B. Kennel, R. Brown, and H. D. I. Abarbanel. Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Phys. Rev. A*, 45(6):3403–3411, 1992.

[28] Dash Associates. *XPRESS-MP User Guide*, 1999. `http://www.dashopt.com`.

[29] N. V. Sahinidis. BARON — Branch And Reduce Optimization Navigator. Technical report, University of Illinois at Urbana-Champaign, Dept. of Chemical Engineering, Urbana, IL, USA, 2000.

[30] A. Bemporad and D. Mignone. *MIQP.M: A Matlab function for solving mixed integer quadratic programs.* ETH Zurich, 2000. code available at `http://control.ethz.ch/~hybrid/miqp`.

[31] ILOG, Inc. *CPLEX 6.5 User Manual.* Gentilly Cedex, France, 1999.

[32] A. Bemporad and M. Morari. Verification of hybrid systems via mathematical programming. In F.W. Vaandrager and J.H. van Schuppen, editors, *Hybrid Systems: Computation and Control*, volume 1569 of *Lecture Notes in Computer Science*, pages 31–45. Springer Verlag, 1999.

[33] A. Bemporad, F.D. Torrisi, and M. Morari. Verification of mixed logical dynamical models — the batch evaporator process benchmark. Technical Report AUT00-04, Automatic Control Laboratory, ETH Zurich, Switzerland, February 2000. `http://control.ethz.ch/`.

A Proof of Lemma 1

Proof. We prove the statement by induction over M. For $M = 0$, we have $f(x) = x - \alpha_0$, so $f^{-1}(y) = y + \alpha_0$. Now suppose that the statement holds for a certain $M - 1 \geq 0$, and consider a function f with M breakpoints. For $x < \alpha_M/\beta_M$, f has only $M - 1$ breakpoints, so in this region f^{-1} can be written as in (20). For $x \geq \alpha_M/\beta_M$, we get

$$f(x) = \left(\sum_{m=0}^{M} s_m\beta_m\right) x - \sum_{m=0}^{M} s_m\alpha_m \ , \tag{25}$$

so

$$f^{-1}(y) = \frac{y + \sum_{m=0}^{M} s_m\alpha_m}{\sum_{m=0}^{M} s_m\beta_m} \ . \tag{26}$$

Since $x = f^{-1}(y)$, we can rewrite the condition $x \geq \alpha_M/\beta_M$ as

$$\begin{aligned}
&\frac{y+\sum_{m=0}^{M} s_m\alpha_m}{\sum_{m=0}^{M} s_m\beta_m} \geq \frac{\alpha_M}{\beta_M} \quad \Leftrightarrow \\
&\Leftrightarrow \quad \frac{1}{\sum_{m=0}^{M} s_m\beta_m}\left(y - \frac{1}{\beta_M}\sum_{m=0}^{M-1} s_m(\beta_m\alpha_M - \beta_M\alpha_m)\right) \geq 0 \\
&\Leftrightarrow \quad y \geq \frac{\tilde{\alpha}_M}{\tilde{\beta}_M}.
\end{aligned} \tag{27}$$

Now, for $y \geq \tilde{\alpha}_M/\tilde{\beta}_M$, by using simple but tedious calculations we can rewrite (26) as

$$\begin{aligned}
f^{-1}(y) = &\frac{y+\sum_{m=0}^{M} s_m\alpha_m}{\sum_{m=0}^{M} s_m\beta_m} + \\
&+ \frac{-s_M\beta_M}{\sum_{m=0}^{k-1} s_m\beta_m \sum_{m=0}^{k} s_m\beta_m}\left(y - \frac{1}{\beta_M}\sum_{m=1}^{M-1} s_m(\beta_m\alpha_M - \beta_M\alpha_m)\right) = \\
&= f_{M-1}^{-1}(y) - s_M(\tilde{\beta}_M y - \tilde{\alpha}_M) \ ,
\end{aligned} \tag{28}$$

where $f_{M-1}^{-1}(y)$ is the inverse of the given function without the Mth max function. Since this function equals $f(x)$ for $x < \alpha_M/\beta_M$, we get $f_{M-1}^{-1}(y) = f^{-1}(y)$ for $y < \tilde{\alpha}_M/\tilde{\beta}_M$. Furthermore, by using a max function and by the induction assumption we now can write

$$\begin{aligned}
f^{-1}(y) &= f_{M-1}^{-1}(y) - s_M \max\{\tilde{\beta}_M y - \tilde{\alpha}_M, 0\} = \\
&= y + \alpha_0 - \sum_{k=1}^{M-1} s_k \max\{\tilde{\beta}_i y - \tilde{\alpha}_i, 0\} - s_M \max\{\tilde{\beta}_M y - \tilde{\alpha}_M, 0\} = \\
&= y + \alpha_0 - \sum_{k=1}^{M-1} s_k \max\{\tilde{\beta}_i y - \tilde{\alpha}_i, 0\} \ ,
\end{aligned} \tag{29}$$

which is just what we wanted to prove. □

B Proof of Proposition 1

Let

$$\begin{aligned}
\xi(t) &= \Phi\xi(t-1) + \mathcal{G}_1 u(t) \\
x(t) &= C\xi(t)
\end{aligned} \tag{30}$$

be a canonical state-space realization of (6a), and consider the output nonlinearity

$$y(t) = C\xi(t) - \bar{\alpha}_0 + \sum_{i=1}^{M} \pm \max\{\bar{\beta}_i C\xi(t) - \bar{\alpha}_i, 0\} \ . \tag{31}$$

Define M auxiliary binary variables

$$[\delta_i(t) = 1] \ \leftrightarrow \ \left[\bar{\beta}_i C\xi(t) - \bar{\alpha}_i \geq 0\right]$$

and M continuous $z_i(t) = (\bar{\beta}_i C\xi(t) - \bar{\alpha}_i)\delta_i(t)$. By using translations into mixed-integer inequalities similar to (12) and (14), the MLD matrices $\mathcal{H}$, $\mathcal{D}_2$, $\mathcal{D}_3$, $\mathcal{E}_2$,…,$\mathcal{E}_5$ can be immediately obtained.