Identification of Deterministic Hybrid ARX Systems via the Identification of Algebraic Varieties*

Yi Ma[1] and René Vidal[2]

[1] Department of Electrical & Computer Engineering, UIUC, Urbana, IL 61801, USA
yima@uiuc.edu
[2] Center for Imaging Science, Johns Hopkins University, Baltimore MD 21218, USA
rvidal@cis.jhu.edu

Abstract. We present a closed-form (linear-algebraic) solution to the identification of hybrid deterministic ARX systems and characterize conditions on the input and switching sequences that guarantee the uniqueness of the solution. We show that the simultaneous identification of the number of ARX systems, the (possibly different) model orders, the ARX model parameters, and the switching sequence is equivalent to the identification and decomposition of a projective algebraic variety whose degree and dimension depend on the number of ARX systems and the model orders, respectively. Given an upper bound for the number of systems, our algorithm identifies the variety and the maximum orders by fitting a polynomial to the data, and the number of systems, the model parameters, and the switching sequence by differentiating this polynomial. Our method is provably correct in the deterministic case, provides a good sub-optimal solution in the stochastic case, and can handle large low-dimensional data sets (up to tens of thousands points) in a batch fashion.

Keywords: System identification, hybrid systems, Auto Regressive eXogenous (ARX) systems, algebraic variety.

1 Introduction

Hybrid systems are mathematical models that are used to describe continuous processes that occasionally exhibit discontinuous behaviors due to sudden changes of dynamics. In recent years, there has been significant interest and progress in the study of the analysis, stability, and control of hybrid systems. Knowing the system parameters, many successful theories have been developed to characterize the behaviors of hybrid systems under different switching mechanisms. However, in practice, the parameters and the switching mechanism of a hybrid system are often not known and we are faced with the task of identifying the system from its input and output measurements.

In this paper, we propose an algebraic approach to the identification of a class of discrete-time hybrid systems known as Hybrid Auto Regressive eXogenous (HARX) systems, i.e., systems whose evolution is described as

$$y_t = \sum_{j=1}^{n_a(\lambda_t)} a_j(\lambda_t) y_{t-j} + \sum_{j=1}^{n_c(\lambda_t)} c_j(\lambda_t) u_{t-j} \quad (+ w_t), \tag{1}$$

where $u_t \in \mathbb{R}$ is the *input* and $y_t \in \mathbb{R}$ is the *output* of the system. The *discrete state* λ_t, also called the *mode* of the system, can evolve due to a variety of switching mechanisms.

* This work is supported by the NSF grant IIS-0347456 and the research startup funds from UIUC Electrical and Computer Engineering Department and Hopkins Whiting School of Engineering. The authors would like to thank professor Robert Fossum for his valuable comments on this paper and Prof. Alexandar Juloski for providing us with the datasets for the experiments.

In this paper, we consider the least restrictive case:[3] $\{\lambda_t\}$ is a deterministic but unknown sequence that can take a finite number of possible values: $\lambda : \mathbb{Z} \to \{1,\ 2,\ \ldots,\ n\}$. The last term w_t is zero for a deterministic hybrid ARX system and is a white-noise random process for a stochastic system. The purpose of this paper is to characterize the (sufficient) conditions and develop efficient algorithms for solving the following problem:

Problem 1 (Identification of Hybrid Auto Regressive eXogenous Systems).

Given the input/output data $\{u_t, y_t\}_{t=0}^{T}$ generated by an HARX system (1), identify the number of constituent systems n, the orders of each ARX system $\{n_a(i), n_c(i)\}_{i=1}^{n}$, the system parameters $\{a_j(i)\}_{j=1}^{n_a(i)}$ and $\{c_j(i)\}_{j=1}^{n_c(i)}$, and the discrete states $\{\lambda_t\}$.

We know from classic identification theory of linear systems that the configuration space of the input/output data generated by a linear system is a subspace whose dimension equals the order of the system. The problem of identifying the system is equivalent to identifying this subspace from a finite number of samples on the subspace. For multiple linear systems, the configuration space is a union of subspaces (possibly of different dimensions), which can be naturally described as a (projective) algebraic variety Z in an ambient space $\mathbb{P}^K$. To some extent, there is a one-to-one correspondence between a hybrid linear system and the variety of its configuration space. Hence the system identification problem can be cast as a special case of the problem of identifying a low-degree (projective) algebraic variety from a finite number of samples. Once the variety Z is known or retrieved from the input/output data, the constituent systems then correspond to the *irreducible sub-varieties* Z_i of the variety Z:

$$Z = Z_1 \cup Z_2 \cup \cdots \cup Z_n \quad \subseteq \mathbb{P}^K.$$

They can be obtained from the *decomposition* of the ideal $\mathfrak{a}(Z)$ of (homogeneous) polynomials associated with the variety Z into prime ideals:

$$\mathfrak{a} = \mathfrak{p}_1 \cap \mathfrak{p}_2 \cap \cdots \cap \mathfrak{p}_n \quad \subseteq \mathbb{C}[z_1, z_2, \ldots, z_K].$$

Our previous work [22] has shown that when the orders of the constituents ARX systems are equal and known, $\mathfrak{a}$ is a principal ideal whose decomposition is equivalent to the factorization of its generator. However, when the orders of the constituent ARX systems are *different*, depending on the switching sequence, the configuration space of the HARX system may not simply be a union of the configuration spaces of the constituent ARX systems, and the ideal $\mathfrak{a}$ is in general *not* a principal ideal. In this paper, we show that the HARX system can still be correctly identified from a special polynomial p whose last nonzero term has the lowest degree-lexicographic order in the ideal $\mathfrak{a}$. This polynomial is unique, factorable, and independent of the switching sequence. The non-repeated factors of this polynomial correspond to the constituent ARX systems, hence the number of systems is given by the number of non-repeated factors.

Relations to Previous Work. Work on identification (and filtering) of hybrid systems first appeared in the seventies; a review of the state of the art as of 1982 can be found in [15]. After a decade-long hiatus, the problem has recently been enjoying considerable interest (see [4, 20, 21] and references therein). Related work has also appeared in the machine

[3] So that our results may apply to other switching mechanisms, such as the Jump-Markov Linear Systems (JMLS) and the PieceWise ARX (PWARX) system, if information about the switching becomes available or needs to be inferred.

learning community (see [7, 14] and references therein). When the model parameters and the switching mechanism (not the discrete states) are *known*, the identification problem reduces to the design of observers for the hybrid state [1, 3, 8, 16], together with the study of observability conditions under which hybrid observers operate correctly (see [20, 21] and references therein). When the model parameters and the switching mechanism are both *unknown*, the identification problem becomes much more challenging. Existing work has concentrated on the class of piecewise affine and piecewise ARX systems, i.e., models in which the regressor space is partitioned into polyhedra with affine or ARX submodels for each polyhedron. For instance, [9] assumes that the number of systems is known, and proposes an identification algorithm that combines clustering, regression and classification techniques; [6] solves for the model parameters and the partition of the state space using mixed-integer linear and quadratic programming; [5] uses a greedy approach for partitioning a set of infeasible inequalities into a minimum number of feasible subsystems, and then iterates between assigning data points to models and computing the model parameters, known as the Expectation Maximization (EM) method. [17] has proposed an alternative solutoin to the identification of both hybrid ARX and state-space models.

Compared to previous work on this topic, this paper makes a few new advancements: 1. It no longer requires the orders of the ARX systems to be known and equal as in [22], or the number of ARX systems to be known as in [19]. 2. It provides a closed-form linear-algebraic algorithm that, given an upper bound for the number of systems and and the system orders, uniquely identifies the number of systems, the system orders, the ARX model parameters, and the switching sequence. Although the algorithm is developed primarily for the noise-free deterministic case, the algorithm is numerically stable and provides a sub-optimal solution for the stochastic case with moderate noises (see Remarks 1 and 4 in the sequel). The solution can be used to initialize other iterative method (such as EM).

2 Identification of a Single ARX System

For the sake of completeness and comparison, let us first review some classic results for the identification of a single discrete-time ARX system

$$y_t = a_1 y_{t-1} + \cdots + a_{n_a} y_{t-n_a} + c_1 u_{t-1} + \cdots + c_{n_c} u_{t-n_c}. \tag{2}$$

The transfer function $\hat{H}(z) \doteq \hat{y}(z)/\hat{u}(z)$ of the system (2) is given by:

$$\hat{H}(z) = z^{\max(n_a-n_c,0)}\tilde{H}(z) = \frac{z^{\max(n_a-n_c,0)}(z^{n_c-1}c_1 + z^{n_c-2}c_2 + \cdots + c_{n_c})}{z^{\max(n_c-n_a,0)}(z^{n_a} - z^{n_a-1}a_1 - z^{n_a-2}a_2 - \cdots - a_{n_a})}. \tag{3}$$

In principle, given infinite input and output sequences, we can identify the parameters of the ARX model by directly computing $\hat{H}(z)$ as $\hat{y}(z)/\hat{u}(z)$.[4] This requires the ARX model to be *identifiable*, i.e., $\tilde{H}(z)$ must have no pole-zero cancellation[5], and $\hat{u}(z)$ to have no zero in common with a pole of $\hat{H}(z)$ and vice versa.

Alternatively, we may identify the system via the identification of a *subspace* associated with the input/output data. Let us define $K \doteq n_a + n_c + 1$ and the vector of *regressors* to be:

$$\boldsymbol{x}_t \doteq \left[y_t, y_{t-1}, \ldots, y_{t-n_a}, u_{t-1}, u_{t-2}, \ldots, u_{t-n_c}\right]^T \quad \in \mathbb{R}^K. \tag{4}$$

[4] Notice that this scheme is not practical since it requires one to obtain the typically infinitely-long output sequence $\{y_t\}$.

[5] That is, the polynomials $z^{\max(n_c-n_a,0)}(z^{n_a} - z^{n_a-1}a_1 - z^{n_a-2}a_2 - \cdots - a_{n_a})$ and $z^{n_c-1}c_1 + z^{n_c-2}c_2 + \cdots + c_{n_c}$ are co-prime.

Thus, for all time t, the so-defined $\boldsymbol{x}_t$ is orthogonal to the vector that consists of the parameters of the ARX system:

$$\boldsymbol{b} \doteq \big[1, -a_1, -a_2, \ldots, -a_{n_a}, -c_1, -c_2, \ldots, -c_{n_c}\big]^T \quad \in \mathbb{R}^K. \tag{5}$$

That is, $\forall t$ $\boldsymbol{x}_t$ and $\boldsymbol{b}$ satisfy the equation $\boldsymbol{b}^T\boldsymbol{x}_t = 0$. In other words, $\boldsymbol{b}$ is the normal vector to the hyperplane spanned by (the rows of) the following *data matrix*:

$$L(n_a, n_c) \doteq [\boldsymbol{x}_{\max(n_a,n_c)}, \ldots, \boldsymbol{x}_{t-1}, \boldsymbol{x}_t, \boldsymbol{x}_{t+1}, \ldots]^T \quad \in \mathbb{R}^{\infty\times K}. \tag{6}$$

When the model orders n_a, n_c are known, we can readily solve for the model parameters $\boldsymbol{b}$ from the null space of $L(n_a, n_c)$. In practice, however, the model orders may be unknown, and only upper bounds $\bar{n}_a$ and $\bar{n}_c$ may be available, hence the vector of regressors $\boldsymbol{x}_t$ is

$$\boldsymbol{x}_t \doteq \big[y_t, y_{t-1}, y_{t-2}, \ldots, y_{t-\bar{n}_a}, u_{t-1}, u_{t-2}, \ldots, u_{t-\bar{n}_c}\big]^T \quad \in \mathbb{R}^K, \tag{7}$$

where $K = \bar{n}_a + \bar{n}_c + 1$. Obviously, the following vector

$$\boldsymbol{b} \doteq \; 1, -a_1, -a_2, \ldots, -a_{n_a}, \mathbf{0}_{1\times(\bar{n}_a-n_a)}, -c_1, -c_2, \ldots, -c_{n_c}, \mathbf{0}_{1\times(\bar{n}_c-n_c)}\;^T \tag{8}$$

satisfies the equation $\boldsymbol{x}_t^T\boldsymbol{b} = 0$ for all t. Notice that here the vector $\boldsymbol{b}$ is the one in (5) with additional $\bar{n}_a - n_a$ and $\bar{n}_c - n_c$ zeros filled in after the terms $-a_{n_a}$ and $-c_{n_c}$, respectively.

Let us define the data matrix $L(\bar{n}_a.\bar{n}_c)$ in the same way as in equation (6). Because of the redundant embedding (7), the vector $\boldsymbol{b}$ is no longer the only one in the null space of L. It is easy to verify that all the following vectors are also in the null space of L:

$$\begin{aligned}
\boldsymbol{b}^1 = \; & \mathbf{0}_{1\times1}, 1, -a_1, \ldots, -a_{n_a}, \mathbf{0}_{1\times(\bar{n}_a-n_a-1)}, \mathbf{0}_{1\times1}, -c_1, \ldots, -c_{n_c}, \mathbf{0}_{1\times(\bar{n}_c-n_c-1)}\;^T, \\
\boldsymbol{b}^2 = \; & \mathbf{0}_{1\times2}, 1, -a_1, \ldots, -a_{n_a}, \mathbf{0}_{1\times(\bar{n}_a-n_a-2)}, \mathbf{0}_{1\times2}, -c_1, \ldots, -c_{n_c}, \mathbf{0}_{1\times(\bar{n}_c-n_c-2)}\;^T, \\
\vdots \qquad & \qquad\qquad\qquad\qquad\qquad\qquad \vdots
\end{aligned} \tag{9}$$

Therefore, the data $\{\boldsymbol{x}_t\}$ span a low-dimensional linear subspace $\mathcal{S}$ in the ambient space $\mathbb{R}^K$.[6] Each of the vectors defined above uniquely determines the original system (2), including its order and coefficients. However, a vector in the null space of L is in general a linear combination of all such vectors and it is not necessarily one of the above. Thus, in order to identify the original system from the data matrix L, we need to seek a vector in its null space that has certain desired structure.

Notice that the last $\bar{n}_c - n_c$ entries of $\boldsymbol{b}$ in (8) are zero, hence the last non-zero entry of $\boldsymbol{b}$ has the lowest order – in terms of the ordering of the entries of $\boldsymbol{x}_t$ – among all vectors that are in the null space of L. Therefore, we can obtain the first $\bar{n}_a + n_c + 1$ entries of $\boldsymbol{b}$ from the null space of the submatrix of L defined by its first $\bar{n}_a + n_c + 1$ columns. Since n_c is unknown, we can incrementally take the first $j = 1, 2, \ldots$ columns of the matrix L from the left to the right:

$$L^1 \doteq L(:, 1:1), \quad L^2 \doteq L(:, 1:2), \quad \ldots, \quad L^j \doteq L(:, 1:j), \tag{10}$$

until the rank of the submatrix L^j stops increasing for the first time for some $j = m$.[7] Under the additional assumption that $\hat{u}(z)$ has no zeros at $z = 0$, the null space of L^m indeed gives the first m entries of the desired vector $\boldsymbol{b}$, because the polynomials $z^{d+\max(n_c-n_a,0)}(z^{n_a} - z^{n_a-1}a_1 - z^{n_a-2}a_2 - \cdots - a_{n_a})$ and $z^{n_c-1}c_1 + z^{n_c-2}c_2 + \cdots + c_{n_c}$ are co-prime.[8]

[6] Only when the initial conditions $\{y_{t_0-1}, \ldots, y_{t_0-\bar{n}_a}\}$ are arbitrary do the data span a hyperplane in $\mathbb{R}^K$ with $\boldsymbol{b}$ as the only normal vector.

[7] If n_c was known, then we would have $m = \bar{n}_a + n_c + 1$.

[8] Similar arguments and conclusions hold if in the definition of $\boldsymbol{x}_t$, we put the inputs $u_{t-1}, \ldots, u_{t-\bar{n}_c}$ in front of the outputs $y_{t-1}, \ldots, y_{t-\bar{n}_a}$ instead.

Remark 1 (Identifying $\boldsymbol{b}$ and m in the Stochastic Case). In the stochastic case (i.e., $w_t \neq 0$), the ultimate goal is to minimize the (squared) modeling error $\sum_t w_t^2 = \sum_t (\boldsymbol{b}^T \boldsymbol{x}_t)^2$, which corresponds to the maximum-likelihood estimate when w_t is white-noise. Then the optimal solution $\boldsymbol{b}^*$ can be found in a least-square sense as the singular vector that corresponds to the smallest singular value of L^m. However, in the noisy case, we cannot directly estimate m from the rank of L^j since it might be full rank for all j. Based on model selection techniques [12], m can be estimated from a noisy L^j as

$$m = \operatorname*{argmin}_{j=1,\ldots,K} \Big\{ \frac{\sigma_j^2(L^j)}{\sum_{k=1}^{j-1} \sigma_k^2(L^j)} + \kappa \cdot j \Big\}, \tag{11}$$

where $\sigma_k(L^j)$ is the kth singular value of L^j and $\kappa \in \mathbb{R}$ is a parameter weighting the two terms. The above criterion minimizes a cost function that consists of a data fitting term and a model complexity term. The data fitting term measures how well the data is approximated by the model – in this case how close the matrix L^j is to dropping rank. The model complexity term penalizes choosing models of high complexity – in this case choosing a large rank.

There is, however, a much more direct way of dealing with the case of unknown orders. The following lemma shows that the system orders n_a and n_c together with the system parameters $\boldsymbol{b}$ can all be simultaneously and uniquely computed from the data.

Lemma 1 (Identifying the Orders of an ARX System). *Suppose we are given data generated by an identifiable ARX model whose input $\hat{u}(z)$ shares no poles or zeros with the zeros or poles, respectively, of the model transfer function $\hat{H}(z)$. If $\bar{n}_a + \bar{n}_c + 1 \leq n_a + n_c + 1$, then*

$$\operatorname{rank}\big(L(\bar{n}_a, \bar{n}_c)\big) = \begin{cases} \bar{n}_a + \bar{n}_c & \text{if and only if} \quad \bar{n}_a = n_a \text{ and } \bar{n}_c = n_c, \\ \bar{n}_a + \bar{n}_c + 1 & \text{otherwise.} \end{cases} \tag{12}$$

Therefore the systems orders can be computed as:

$$(n_a, n_c) = \operatorname*{arg\,min}_{(\bar{n}_a, \bar{n}_c) \in \mathbb{Z}^2} \{\bar{n}_a + \bar{n}_c : \operatorname{rank}(L(\bar{n}_a, \bar{n}_c)) = \bar{n}_a + \bar{n}_c\}. \tag{13}$$

The parameter vector $\boldsymbol{b}$ is the unique vector in the null space of $L(n_a, n_c)$.

We omit the proof here due to the limit of space. In principle, the lemma allows us to identify the precise orders n_a, n_c and the vector $\boldsymbol{b}$ of the ARX system from the (infinite) sequences of input $\{u_t\}$ and output $\{y_t\}$. In practice, we are usually given a finite input/output sequence. In such cases, we need to assume that the sequence of regressors is *sufficiently exciting*, i.e., the $T \times (n_a + n_c + 1)$ submatrix

$$L \doteq [\boldsymbol{x}_{\max(n_a,n_c)}, \ldots, \boldsymbol{x}_{\max(n_a,n_c)+T-1}]^T$$

has the same rank $n_a + n_c$ as the "full" L matrix defined in (6). [9] This condition for identifiability from finite data can also be expressed in terms of the input sequence. As shown in [2], the regressors are sufficiently exciting if the input sequence $\{u_t\}$ is, i.e., if the following vectors

$$\boldsymbol{u}_t \doteq [u_t, u_{t-1}, \ldots, u_{t-n_a-n_c+1}]^T \in \mathbb{R}^{n_a+n_c}, \qquad n_a + n_c - 1 \leq t \leq T,$$

span an $(n_a + n_c)$-dimensional subspace.

[9] In the case of a redundant embedding, the sequence of regressors is said to be sufficiently exciting if $\operatorname{rank}(L) = \bar{n}_a + n_c + 1$.

3 Identification of Hybrid ARX Systems

From our discussion in the previous section, we know that the regressors generated by an identifiable ARX system with sufficiently exciting input live in a linear subspace in $\mathbb{R}^K$ where $K = \bar{n}_a + \bar{n}_c + 1$ and $\bar{n}_a, \bar{n}_c$ are upper bounds on the orders of the system. The problem of identifying the ARX system becomes one of seeking a vector in the orthogonal complement to this subspace that has certain desired structure. We show in this section how to generalize these concepts to the more challenging problem of identifying a hybrid ARX system (Problem 1).

Consider an input/output sequence $\{u_t, y_t\}$ generated by a hybrid ARX system switching among a set of n ARX systems with parameters $\{\boldsymbol{b}_i\}_{i=1}^n$ and possibly different orders $\{n_a(i), n_c(i)\}_{i=1}^n$. We assume that the HARX system is *identifiable*, i.e., for all $i = 1, \ldots, n$, the rational function $\tilde{H}_i(z)$ associated with the ith ARX model has no zero-pole cancellation and the configuration subspaces of all the ARX models do not contain one another.[10] In general, we also assume that we do not know the exact orders of the systems but know only certain upper bounds of them, i.e.,

$$\bar{n}_a \geq n_a \doteq \max\{n_a(1), \ldots, n_a(n)\}, \quad \bar{n}_c \geq n_c \doteq \max\{n_c(1), \ldots, n_c(n)\}.$$

Very often we do not know the exact number of systems involved either but know only an upper bound of it, i.e., $\bar{n} \geq n$.[11] In this section, we study how to identify such a hybrid ARX system despite these uncertainties.

3.1 The Hybrid Decoupling Polynomial

One of the difficulties in identifying hybrid ARX systems is that we do not know the switching sequence λ_t, hence we cannot directly apply the subspace identification technique described in the previous section to each of the n ARX systems. As we will soon see, in fact both the number of subspaces and their dimensions depend not only on the number of systems and their orders but also on the switching sequence. This motivates us to look for relationships between the data $\{\boldsymbol{x}_t \in \mathbb{R}^K\}$ and the system parameters $\{\boldsymbol{b}_i \in \mathbb{R}^K\}$ that do not depend on the switching sequence. To this end, recall that for every t there exists a state $\lambda_t = i \in \{1, 2, \ldots, n\}$ such that $\boldsymbol{b}_i^T \boldsymbol{x}_t = 0$. Therefore, the following polynomial equation [22] must be satisfied by the system parameters and the input/output data for any switching sequence and mechanism (JMLS or PWARX):

$$p_n(\boldsymbol{x}_t) \doteq \prod_{i=1}^{n} (\boldsymbol{b}_i^T \boldsymbol{x}_t) = 0. \tag{14}$$

We call this polynomial equation the *hybrid decoupling polynomial* (HDP). In the absence of knowledge about the switching mechanism, the HDP encodes all the information about the system parameters that we can obtain from the input/output data.

The HDP eliminates the discrete state by taking the product of the equations defining each one of the ARX systems. While taking the product is not the only way of algebraically eliminating the discrete state, this leads to an algebraic equation with a very nice

[10] One way to ensure this is to assume that for all $i \neq j = 1, \ldots, n$, $\tilde{H}_i(z)$ and $\tilde{H}_j(z)$ do not have all their zeros and poles in common. That is, there is no ARX system that can simulate another ARX system with a smaller order. However, this is not necessary because two ARX systems can have different configuration spaces even if one system's zeros and poles are a subset of the other's.

[11] This is the case when a particular switching sequence visits only a subset of all the discrete states.

algebraic structure. The HDP is simply a homogeneous multivariate polynomial of degree n in K variables

$$p_n(\boldsymbol{z}) \doteq \prod_{i=1}^{n} \left(\boldsymbol{b}_i^T \boldsymbol{z}\right) = 0, \tag{15}$$

which can be written linearly in terms of its coefficients as

$$p_n(\boldsymbol{z}) \doteq \sum h_{n_1,\ldots,n_K} z_1^{n_1} \cdots z_K^{n_K} = \boldsymbol{h}_n^T \nu_n(\boldsymbol{z}) = 0. \tag{16}$$

In eqn. (16), $h_I = h_{n_1,\ldots,n_K} \in \mathbb{R}$ is the coefficient of the monomial $\boldsymbol{z}^I = z_1^{n_1} z_2^{n_2} \cdots z_K^{n_K}$, where $0 \leq n_j \leq n$ for $j = 1, \ldots, K$, and $n_1 + n_2 + \cdots + n_K = n$; $\nu_n : \mathbb{R}^K \to \mathbb{R}^{M_n(K)}$ is the *Veronese map* of degree n which is defined as [10]:

$$\nu_n : \ [z_1, \ldots, z_K]^T \mapsto [\ldots, \boldsymbol{z}^I, \ldots]^T, \tag{17}$$

with I chosen in the degree-lexicographic order (assuming the order $z_1 < z_2 < \cdots < z_K$); and $M_n(K) \doteq \binom{n+K-1}{K-1} = \binom{n+K-1}{n}$ is the total number of independent monomials. As shown in [10], the vector of coefficients $\boldsymbol{h}_n \in \mathbb{R}^{M_n(K)}$ is simply a vector representation of the symmetric tensor product of the individual system parameters $\{\boldsymbol{b}_i\}_{i=1}^n$, i.e.,

$$\text{Sym}(\boldsymbol{b}_1 \otimes \boldsymbol{b}_2 \otimes \cdots \otimes \boldsymbol{b}_n) \doteq \sum_{\sigma \in \mathfrak{S}_n} \boldsymbol{b}_{\sigma(1)} \otimes \boldsymbol{b}_{\sigma(2)} \otimes \cdots \otimes \boldsymbol{b}_{\sigma(n)} \ \in \mathbb{R}^{M_n(K)}, \tag{18}$$

where $\mathfrak{S}_n$ is the permutation group of n elements. We will show in the sequel how this vector can be recovered from the data and how the parameters of each individual ARX system can be further retrieved from it.

3.2 Identifying the Number and Orders of ARX Systems

Let us assume for now that we know the number of systems n. We will show later how to relax this assumption. Since the HDP (14) – (16) is satisfied by all the data points $\{\boldsymbol{x}_t\}_{t=1}^T$, we can use it to derive the following linear system on the vector $\boldsymbol{h}_n$:

$$L_n(\bar{n}_a, \bar{n}_c)\ \boldsymbol{h}_n \doteq \begin{bmatrix} \nu_n(\boldsymbol{x}_{\max\{\bar{n}_a,\bar{n}_c\}})^T \\ \nu_n(\boldsymbol{x}_{\max\{\bar{n}_a,\bar{n}_c\}+1})^T \\ \vdots \\ \nu_n(\boldsymbol{x}_{\max\{\bar{n}_a,\bar{n}_c\}+T-1})^T \end{bmatrix} \boldsymbol{h}_n = \boldsymbol{0}_{T\times 1}, \tag{19}$$

where $L_n(\bar{n}_a, \bar{n}_c) \in \mathbb{R}^{T \times M_n(K)}$ is the matrix of the input/output data embedded via the Veronese map.

Notice that to construct the matrix L_n, one needs to choose $\bar{n}_a$ and $\bar{n}_c$. If the constituent ARX systems have different orders, the choice can never be the most compact for every ARX system. Nevertheless, there will always be less redundancy in the embedding if $\bar{n}_a$, $\bar{n}_c$ are the maximum orders n_a, n_c for all the ARX systems. To identify the maximum orders, we need some extra conditions on the switching and input sequences.

Definition 1 (Sufficiently Exciting Switching and Input Sequences). *A switching and input sequence $\{\lambda_t, u_t\}$ is called* sufficiently exciting *for a hybrid ARX system, if the data points $\{\boldsymbol{x}_t\}$ generated by $\{\lambda_t, u_t\}$ are sufficient to determine the union of the subspaces associated with the constituent ARX systems as an algebraic variety.*

Remark 2. When $\bar{n}_a < n_a$ or $\bar{n}_c < n_c$, the above condition requires $L_n(\bar{n}_a, \bar{n}_c)$ to be full rank, because at least one of the subspaces must have full dimension $\bar{n}_a + \bar{n}_c + 1$. When $\bar{n}_a \geq n_a$ and $\bar{n}_c \geq n_c$, the above condition implies that the null space of $L_n(\bar{n}_a, \bar{n}_c)$ is contained in the span of the vectors $\{\boldsymbol{h}\}$, where $\boldsymbol{h}$ is the symmetric tensor product of any choice of n vectors of the form (8) or (9), each one associated with one of the n ARX models.

Remark 3. The above condition is not as strong as it seems to be, as the set of input and switching sequences that are not sufficiently exciting are a zero-measure set. Notice, however, that the definition does not explicitly characterize the set of sufficiently exciting input and switching sequences. Intuitively the switching sequence should visit each one of the n modes frequently enough and the input sequence should be sufficiently exciting, as defined in the previous section. A more precise characterization of sufficiently exciting input and switching sequences remains elusive at this point.

Thanks to the above definition of sufficiently exiting input and switching sequences, the following theorem gives a formula for the maximum orders. The theorem is a natural generalization of Lemma 1 for a single ARX system to the case of hybrid ARX systems.

Theorem 1 (Identifying the Maximum Orders). *Let $\{u_t, y_t\}$ be the input/output data generated by an identifiable HARX system. Let $L_n(i,j) \in \mathbb{R}^{T \times M_n(i+j+1)}$ be the embedded data matrix defined in (19), but computed with system orders i and j. If T is large enough and the input and switching sequences are sufficiently exciting, then the maximum orders of the constituent ARX systems are given by:*

$$(n_a, n_c) = \underset{(i,j):M_n(i+j+1)<T}{\arg\min} \big\{(i+j) : \text{rank}\big(L_n(i,j)\big) < M_n(i+j+1)\big\}. \quad (20)$$

Proof. First notice that the maximum orders n_a and n_c maybe achieved separately by different ARX systems. Nevertheless, for any ARX system, if either $i < n_a$ or $j < n_c$ is true, at least one of the subspaces must be of dimension $i + j + 1$. Therefore if the input and switching sequences are sufficiently exciting so that this subspace is visited enough, then there is a large enough T such that the entries of $L_n(i,j)$ are independent monomials of degree n on these regressors. The matrix $L_n(i,j)$ drops rank only for a zero measure set of such regressors. Therefore in general, for a sufficiently large T, there is no polynomial of degree n that vanishes on the set of all regressors and we must have $\text{rank}(L_n(i,j)) = M_n(i+j+1)$. If $i = n_a$ and $j = n_c$, then there is exactly one vector, i.e., $\boldsymbol{h}_n$, in the null space of $L_n(i,j)$. Therefore, the maximum orders n_a, n_c are the ones for which $n_a + n_c$ is minimum and $L_n(n_a, n_c)$ drops rank, as claimed. ∎

Given the data matrix $L_n(n_a, n_c)$ embedded with the correct maximum orders, we would like to retrieve the coefficient vector $\boldsymbol{h}_n$ from its null space. There are two potential difficulties. First, since the maximum orders n_a, n_c may not be tight for every constituent ARX system, the null space of $L_n(n_a, n_c)$ may be more than one-dimensional, as we have known from a single ARX system. Second, even if we know the discrete state for each time, the structure of the data associated with each state is not exactly the same as that of the ARX system itself: Suppose we switch to the ith system at time t_0, then we have $\boldsymbol{b}_i^T \boldsymbol{x}_{t_0} = 0$. However, the vectors $\boldsymbol{b}$ given in equation (9) are no longer orthogonal to $\boldsymbol{x}_{t_0}$ even if the embedding is redundant for the ith system. In a sense, the regressor at a switching time usually lives in a subspace whose dimension is higher than that of the subspace associated with the ARX model generating the regressor. Therefore, the configuration space of the data $\{\boldsymbol{x}_t\}$ of an HARX system will *not* exactly be the union of all the subspaces associated with the constituent ARX systems. Let us denote the former as an algebraic variety Z' and the latter as Z. Then in general, we have $Z' \supseteq Z$.

In order to retrieve $\boldsymbol{h}_n$ uniquely from the data matrix L_n, we need to utilize its additional structure.

Lemma 2 (Structure of the Hybrid Decoupling Polynomial). *The monomial associated with the last non-zero entry of the coefficient vector $\boldsymbol{h}_n$ of the hybrid decoupling polynomial $p_n(\boldsymbol{z}) = \boldsymbol{h}_n^T \nu_n(\boldsymbol{z})$ has the lowest degree-lexicographic order in all the polynomials in $\mathfrak{a}(Z) \cap S_n$.*[12]

Proof. Any polynomial of degree n in the ideal $\mathfrak{a}(Z)$ is a superposition of $\prod_{i=1}^{n}(\boldsymbol{b}_{\sigma(i)}^T \boldsymbol{z})$ where $\boldsymbol{b}_{\sigma(i)}$ is a normal vector to the subspace associated with the ith ARX system.[13] Notice that $\boldsymbol{h}_n$ is the symmetric tensor of $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n$ defined in (8). For the ith ARX system, the last non-zero entry of the vector $\boldsymbol{b}_i$ always has the lowest degree-lexicographic order among all normal vectors that are orthogonal to the regressors $\boldsymbol{z} = \boldsymbol{x}_t$ associated to the ith system, see equations (8) and (9). Therefore, the last non-zero entry of $\boldsymbol{h}_n$ must have the lowest degree-lexicographic order. ∎

Theorem 2 (Identifying the Hybrid Decoupling Polynomial). *Let $\{u_t, y_t\}_{t=0}^{T}$ be the input/output data generated by an identifiable HARX system. Let $L_n^j \in \mathbb{R}^{T \times j}$ be the first j columns of the embedded data matrix $L_n(n_a, n_c)$, and let*

$$m \doteq \min\big\{j : \text{rank}\big(L_n^j\big) = j - 1\big\}. \tag{21}$$

If T is sufficiently large and the input and switching sequences are sufficiently exciting, then the coefficient vector $\boldsymbol{h}_n$ of the hybrid decoupling polynomial is given by

$$\boldsymbol{h}_n = \big[\big(\boldsymbol{h}_n^m\big)^T,\ \boldsymbol{0}_{1\times(M_n(K)-m)}\big]^T \quad \in \mathbb{R}^{M_n(K)}, \tag{22}$$

where $\boldsymbol{h}_n^m \in \mathbb{R}^m$ is the unique vector that satisfies

$$L_n^m \boldsymbol{h}_n^m = \boldsymbol{0} \quad \text{and} \quad \boldsymbol{h}_n^m(1) = 1. \tag{23}$$

Proof. Let Z to be the union of the subspaces associated with the n constituent ARX systems. Since the input and switching sequence is sufficiently exciting in the sense of Definition 1, any polynomial of degree less than and equal to n that vanishes on all the data points must be in the set $\mathfrak{a}(Z) \cap S^n$.[14]

From our discussion before the theorem, the configuration space Z' of the data $\{\boldsymbol{x}_t\}$ associated with the hybrid ARX system is in general a superset of Z. The ideal $\mathfrak{a}'(Z')$ of polynomials that vanish on the configuration space Z' is then a sub-ideal of the ideal $\mathfrak{a}(Z)$ associated with the union of the subspaces. Furthermore, regardless of the switching sequence, the hybrid decoupling polynomial $p_n(\boldsymbol{z})$ is always in $\mathfrak{a}' \cap S_n \subseteq \mathfrak{a} \cap S_n$. According to Lemma 2, the last non-zero term of $p_n(\boldsymbol{z})$ has the lowest degree-lexicographic order among all polynomials of degree n in $\mathfrak{a}$, so does it in $\mathfrak{a}'$. Since every solution $L_n \tilde{\boldsymbol{h}} = 0$ gives a polynomial $\tilde{p}_n(\boldsymbol{z}) = \tilde{\boldsymbol{h}}_n^T \nu_n(\boldsymbol{z}) \in \mathfrak{a} \cap S_n$ of degree n that vanishes on all data points, the last non-zero entry of $\boldsymbol{h}_n$ given by (22) obviously has the lowest degree-lexicographic order. Therefore, we have $p_n(\boldsymbol{z}) = \boldsymbol{h}_n^T \nu_n(\boldsymbol{z})$. ∎

In fact to compute the coefficients $\boldsymbol{h}_n$ of the hybrid decoupling polynomial, we can do better than checking the rank of the submatrix L_n^j for every $j = 1, 2, \ldots$. The following corollary provides one alternative scheme.

Corollary 1 (Zero Coefficients of the Decoupling Polynomial). *Consider a set of vectors $\boldsymbol{b}_i \in \mathbb{R}^K, i = 1, \ldots, n$. Suppose that one of the $\boldsymbol{b}_i$ has a maximal number of zeros on its right, and without loss of generality, assume it is $\boldsymbol{b}_1 = [b_{11}, b_{12}, \ldots, b_{1n_1}, 0, \ldots, 0]^T$,*

[12] The set of (homogeneous) polynomial of degree n.

[13] This is easily verifiable from the fact that the derivatives of the polynomials in $\mathfrak{a}(Z)$ are exactly the normal vectors of the subspaces.

[14] S^n is the set of polynomials of degree up to n.

with $b_{1n_1} \neq 0$. The multivariate polynomial $p_n(\boldsymbol{z}) \doteq (\boldsymbol{b}_1^T\boldsymbol{z})(\boldsymbol{b}_2^T\boldsymbol{z})\cdots(\boldsymbol{b}_n^T\boldsymbol{z})$ has zero coefficients for all the monomials of $\nu_n\big([z_{n_1+1}, z_{n_1+2}, \ldots, z_K]\big)$; but the coefficients cannot all be zeros for the monomials of $\nu_n\big([z_{n_1}, z_{n_1+1}, \ldots, z_K]\big)$.

This corollary allows us to narrow down the range for m (where L_n^j first drops rank) because m must fall between two consecutive values of the following: $1,\ M_n(K) - M_n(K-1),\ M_n(K) - M_n(K-2),\ \ldots,\ M_n(K) - 1$.

Remark 4 (Sub-Optimality in the Stochastic Case). In the stochastic case (i.e., $w_t \neq 0$), we can still solve for $\boldsymbol{h}_n^m$ in (23) in a least-squares sense as the singular vector of L_n^m associated with its smallest singular value, using a similar model selection criterion for m as in Remark 1. However, unlike the single system case, the so-found $\boldsymbol{h}_n$ no longer minimizes the sum of least-square errors $\sum_t w_t^2 = \sum_t (\boldsymbol{b}_{\lambda_t}^T \boldsymbol{x}_t)^2$. Instead, it minimizes (in a least-square sense) a "weighted version" of this objective:

$$\sum_t \alpha_t (\boldsymbol{b}_{\lambda_t}^T \boldsymbol{x}_t)^2 \doteq \sum_t \prod_{i \neq \lambda_t} (\boldsymbol{b}_i^T \boldsymbol{x}_t)^2 (\boldsymbol{b}_{\lambda_t}^T \boldsymbol{x}_t)^2, \tag{24}$$

where the weight α_t is conveniently chosen to be $\prod_{i\neq\lambda_t}(\boldsymbol{b}_i^T\boldsymbol{x}_t)^2$. Such a "softening" of the objective function allows a global algebraic solution. It offers a sub-optimal approximation for the original stochastic objective when the variance of w_t is small. One can use the solution as the initialization for any other iterative optimization scheme (such as EM) to further minimize the original stochastic objective.

Notice that in the above theorem, we have assumed that the switching sequence is such that all the ARX systems are sufficiently visited. What if only a subset of the n systems are sufficiently visited? Furthermore, in practice, we sometimes do not even know the correct number of systems involved and only know an upper bound for it. The question is whether the above theorem still applies when the degree n we choose for the Veronese embedding is strictly larger than the actually number of systems. This is answered by the following corollary whose proof is straightforward.

Corollary 2 (Identifying the Number of ARX Systems). *Let $\{u_t, y_t\}_{t=0}^T$ be the input/output data generated by an HARX system with $n < \bar{n}$ discrete states. If T is sufficiently large and the input and switching sequences are sufficiently exciting, then the vector $\boldsymbol{h}_{\bar{n}}$ found by Theorem 2 is the symmetric tensor product $\boldsymbol{h}_{\bar{n}} = \mathrm{Sym}\big(\boldsymbol{b}_1 \otimes \boldsymbol{b}_2 \cdots \otimes \boldsymbol{b}_n \otimes \underbrace{\boldsymbol{e}_1 \otimes \cdots \otimes \boldsymbol{e}_1}_{\bar{n}-n}\big)$, where $\boldsymbol{e}_1 \doteq [1, 0, \ldots, 0]^T \in \mathbb{R}^K$, i.e., $\boldsymbol{h}_{\bar{n}}$ is the coefficients of the polynomial $p_{\bar{n}}(\boldsymbol{z}) = \boldsymbol{h}_{\bar{n}}^T \nu_{\bar{n}}(\boldsymbol{z}) = \big(\boldsymbol{b}_1^T\boldsymbol{z}\big)\big(\boldsymbol{b}_2^T\boldsymbol{z}\big)\cdots\big(\boldsymbol{b}_n^T\boldsymbol{z}\big) z_1^{\bar{n}-n}$.*

Therefore, even if we may over-estimate the number of constituent systems or the switching sequence does not visit all the systems, the solution given by Theorem 2 will simply treat the nonexistent (or not visited) systems as if they had zero order[15] and the information about the rest of the systems will be conveniently recovered.

3.3 Identifying the System Parameters and Discrete States

Theorem 2 allows us to determine the hybrid decoupling polynomial $p_n(\boldsymbol{z}) = \boldsymbol{h}_n^T \nu_n(z)$, from input/output data $\{u_t, y_t\}_{t=0}^T$. The rest of the problem is to recover the system parameters $\{\boldsymbol{b}_i\}_{i=1}^n$ from $\boldsymbol{h}_n$. To this end, recall from [22] that given $\boldsymbol{h}_n$ one can recover the

[15] That is, the coefficient vector $\boldsymbol{b} = \boldsymbol{e}_1$ corresponds to the "system" $y_t = 0$ with $n_a = n_c = 0$, which is a trivial ARX system.

model parameters by looking at the partial derivative of $p_n(\boldsymbol{z})$ given in (15)

$$D p_n(\boldsymbol{z}) \doteq \frac{\partial p_n(\boldsymbol{z})}{\partial \boldsymbol{z}} = \sum_{i=1}^{n} \prod_{\ell \neq i} (\boldsymbol{b}_\ell^T \boldsymbol{z}) \boldsymbol{b}_i. \tag{25}$$

If $\boldsymbol{z}$ belongs to hyperplane $\mathcal{H}_i = \{\boldsymbol{z} : \boldsymbol{b}_i^T \boldsymbol{z} = 0\}$, then, since the 1st entry of $\boldsymbol{b}_i$ is equal to one, after replacing $\boldsymbol{b}_i^T \boldsymbol{z} = 0$ into (25) we obtain

$$\boldsymbol{b}_i = \left. \frac{D p_n(\boldsymbol{z})}{\boldsymbol{e}_1^T D p_n(\boldsymbol{z})} \right|_{z \in \mathcal{H}_i} \in \mathbb{R}^K, \tag{26}$$

where $\boldsymbol{e}_1 = [1, 0, \ldots, 0]^T \in \mathbb{R}^K$. Therefore, we can estimate the system parameters directly from the derivatives of $p_n(\boldsymbol{z})$ at a collection of n points $\{\boldsymbol{z}_i \in \mathcal{H}_i\}_{i=1}^n$ lying on the n hyperplanes.

However, since the value of the discrete state λ_t is unknown, we do not know which data points $\{\boldsymbol{x}_t\}_{t=n_a}^T$ belong to which hyperplane. In order to find the set of points $\{\boldsymbol{z}_i \in \mathcal{H}_i\}_{i=1}^n$, let us consider a line with base point $\boldsymbol{z}_0$ and direction $\boldsymbol{v}$, $\mathcal{L} = \{\boldsymbol{z}_0 + \alpha \boldsymbol{v}, \alpha \in \mathbb{R}\}$. If $\boldsymbol{z}_0 \neq 0$, $\boldsymbol{z}_0$ is not parallel to $\boldsymbol{v}$, and $\boldsymbol{b}_i^T \boldsymbol{v} \neq 0$, then the line $\mathcal{L}$ must intersect the union of all hyperplanes $\cup_{i=1}^n \mathcal{H}_i = \{\boldsymbol{z} : p_n(\boldsymbol{z}) = 0\}$ at n distinct points $\boldsymbol{z}_i = z_0 + \alpha_i \boldsymbol{v} \in \mathcal{H}_i \cap \mathcal{L}, i = 1, \ldots, n$, where $\{\alpha_i\}$ are the roots of the univariate polynomial

$$q_n(\alpha) = p_n(\boldsymbol{z}_0 + \alpha \boldsymbol{v}). \tag{27}$$

We are left with choosing the parameters $\boldsymbol{x}_0$ and $\boldsymbol{v}$ for the line $\mathcal{L}$. The base point $\boldsymbol{x}_0$ can be chosen as any nonzero vector in $\mathbb{R}^K$. Given $\boldsymbol{z}_0$, the direction $\boldsymbol{v}$ must be chosen not parallel to $\boldsymbol{z}_0$ and such that $\boldsymbol{b}_i^T \boldsymbol{v} \neq 0$, for all $i = 1, \ldots, n$. Since the latter constraint is equivalent to $p_n(\boldsymbol{v}) \neq 0$, and p_n is known, we can immediately choose $\boldsymbol{v}$ even though we do not know the system parameters $\{\boldsymbol{b}_i\}_{i=1}^n$.

Be aware that if we have chosen for the Veronese embedding a number $\bar{n}$ that is strictly larger than n, the polynomial $p_{\bar{n}}(\boldsymbol{z})$ will be of the form $(\boldsymbol{b}_1^T \boldsymbol{z})(\boldsymbol{b}_2^T \boldsymbol{z}) \cdots (\boldsymbol{b}_n^T \boldsymbol{z}) z_1^{\bar{n}-n}$. Then the line $\mathcal{L}$ will have only $n+1$ intersections with the n hyperplanes $\mathcal{H}_1, \ldots, \mathcal{H}_n$ and the hyperplane $\mathcal{H}_0 \doteq \{\boldsymbol{z} : \boldsymbol{e}_1^T \boldsymbol{z} = z_1 = 0\}$. The intersection $\boldsymbol{z}_0 = \mathcal{H}_0 \cap \mathcal{L}$ has a multiplicity of $\bar{n} - n$; and $D p_{\bar{n}}(\boldsymbol{z}_0) \sim \boldsymbol{e}_1$ if $\bar{n} - n = 1$ or $D p_{\bar{n}}(\boldsymbol{z}_0) = 0$ if $\bar{n} - n > 1$. We have essentially proven the following theorem.

Theorem 3 (Identifying the Constituent System Parameters). *Given the input/output data $\{u_t, y_t\}_{t=0}^T$ generated by an HARX system with n discrete states, the system parameters $\{\boldsymbol{b}_i\}_{i=1}^n$ can be computed from the the hybrid decoupling polynomial $p_{\bar{n}}(\boldsymbol{z}) = \boldsymbol{h}_{\bar{n}}^T \nu_{\bar{n}}(\boldsymbol{z})$ for any $\bar{n} \geq n$ as follows:*

1. *Choose $\boldsymbol{z}_0 \neq 0$ and $\boldsymbol{v}$ such that $\boldsymbol{v} \neq \gamma \boldsymbol{z}_0$ and $p_{\bar{n}}(\boldsymbol{v}) \neq 0$.*
2. *Solve for the $\bar{n}$ roots $\{\alpha_i\}_{i=1}^{\bar{n}}$ of $q_{\bar{n}}(\alpha) = p_{\bar{n}}(\boldsymbol{z}_0 + \alpha \boldsymbol{v}) = 0$.*
3. *For all the roots $\boldsymbol{z}_i = \boldsymbol{z}_0 + \alpha_i \boldsymbol{v}$ with $z_1 \neq 0$, compute the system parameters as $\boldsymbol{b}_i = \frac{D p_{\bar{n}}(\boldsymbol{z}_i)}{\boldsymbol{e}_1^T D p_{\bar{n}}(\boldsymbol{z}_i)} \in \mathbb{R}^K, i = 1, 2, \ldots, n$.*

Remark 5 (Alternative Ways of Identifying $\{\boldsymbol{b}_i\}_{i=1}^n$ from Noisy Data). In the presence of noise, we can still estimate the normal vectors $\{\boldsymbol{b}_i\}_{i=1}^n$ as in Theorem 3. However, the quality of the estimates will depend on the choice of the parameters $\boldsymbol{z}_0$ and $\boldsymbol{v}$. In this case, one can choose multiple $(\boldsymbol{z}_0, \boldsymbol{v})$ satisfying the above conditions, obtain the system parameters for each choice, and let $\{\boldsymbol{b}_i\}_{i=1}^n$ be the parameters that better reconstruct $\boldsymbol{h}_n$. Alternatively, one can directly choose $\{\boldsymbol{z}_i\}_{i=1}^n$ from points in the data set that fit the decoupling polynomial in an optimal way. That allows us to bypass the problem of solving the (real) roots of the real polynomial $q_n(\alpha)$. We refer the interested reader to [18] for further details.

Once the system parameters $\{\boldsymbol{b}_i\}_{i=1}^n$ are recovered, we can then reconstruct the discrete state trajectory $\{\lambda_t\}$ from input/output data $\{\boldsymbol{x}_t\}_{t=0}^T$ as

$$\lambda_t = \arg\min_{i=1,\ldots,n} \left(\boldsymbol{b}_i^T \boldsymbol{x}_t\right)^2, \tag{28}$$

because for each time t there exists a generally unique i such that $\boldsymbol{b}_i^T \boldsymbol{x}_t = 0$. There will be ambiguity in the value of λ_t only if $\boldsymbol{x}_t$ happens to be at (or close to) the intersection of more than one subspace associated to the constituent ARX systems. However, the set of all such points is a zero measure set of the variety $Z \subseteq \{\boldsymbol{z} : p_n(\boldsymbol{z}) = 0\}$.

3.4 The Basic Algorithm and its Extensions

Based on the results that we have derived so far, we summarize the main steps for solving the identification of an HARX system (Problem 1) as the following Algorithm 1. Notice that the algorithm is different from the one proposed in [19] which only deals with the special case with a known number of ARX systems with the same order(s).

Algorithm 1 (Identification of HARX Systems).

Given the input/output data $\{y_t, u_t\}$ from a sufficiently excited hybrid ARX system, and the upper bounds on the number $\bar{n}$ and orders $(\bar{n}_a, \bar{n}_c)$ of its constituent ARX systems:

1. **Maximum System Orders.** Identify the maximum orders (n_a, n_c) according to Theorem 1.
2. **Veronese Embedding.** Construct the data matrix $L_{\bar{n}}(n_a, n_c)$ via the Veronese map (17) based on the given number $\bar{n}$ of systems and the maximum orders (n_a, n_c) identified from the previous step.
3. **Hybrid Decoupling Polynomial.** Compute the coefficients of the polynomial $p_{\bar{n}}(\boldsymbol{z}) \doteq \boldsymbol{h}_{\bar{n}}^T \nu_{\bar{n}}(\boldsymbol{z}) = \prod_{i=1}^n \boldsymbol{b}_i^T \boldsymbol{z}\ z_1^{\bar{n}-n} = 0$ from the data matrix $L_{\bar{n}}$ according to Theorem 2 and Corollary 2. In the stochastic case, comply with Remarks 1 and 4.
4. **Constituent System Parameters.** Retrieve the parameters $\{\boldsymbol{b}_i\}_{i=1}^n$ of each constituent ARX system from $p_{\bar{n}}(\boldsymbol{z})$ according to Theorem 3. In the noisy case, comply with Remark 5.
5. **Key System Parameters.** The correct number of system n is the number of $\boldsymbol{b}_i \neq \boldsymbol{e}_1$; The correct orders $n_a(i), n_c(i)$ are determined from such $\boldsymbol{b}_i$ according to their definition (8); The discrete state λ_t for each time t is given by equation (28).

Different Embedding Orders. The ordering of $\{y_t\}$ and $\{u_t\}$ in (7) is more efficient for the algorithm when $n_a(i)$ are approximately the same for all the constituent systems and $n_c(i)$ are much smaller than $n_a(i)$. However, if $n_a(i)$ are rather different for different systems and $n_c(i)$ and $n_a(i)$ are roughly the same, the following ordering in time t

$$\boldsymbol{x}_t \doteq \left[y_t, y_{t-1}, u_{t-1}, y_{t-2}, u_{t-2}, \ldots, y_{t-n_a}, u_{t-n_a}\right]^T \quad \in \mathbb{R}^K \tag{29}$$

results in less non-zero leading coefficients in $\boldsymbol{h}_n$, thus making Algorithm 1 more efficient. However, if all the systems have the same $n_a = n_c$, both embeddings have the same efficiency.

Inferring the Switching Mechanisms. Once the system parameters and the discrete state have been identified, the problem of estimating the switching mechanisms, e.g., the partition of the state space for PWARX or the parameters of the jump Markov process for JMLS, becomes a simpler problem. We refer interested readers to [5, 9] for specific algorithms.

4 Simulations and Experiments

In this section we evaluate the performance of the proposed algorithm with respect to the amount of noise in the data and the choice of the model orders. We also present experiments on real data from a component placement process in a pick-and-place machine.

Error as a Function of Noise. Consider the PWAR model taken from [13]

$$y_t = \begin{cases} 2u_{t-1} + 10 + w_t & \text{if} \quad u_{t-1} \in [-10, 0], \\ -1.5u_{t-1} + 10 + w_t & \text{if} \quad u_{t-1} \in (0, 10], \end{cases} \tag{30}$$

with input $u_t \overset{\text{i.i.d.}}{\sim} \mathcal{U}(-10, 10)$ and noise $w_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_\eta^2)$. We run our algorithm with $n = 2$, $n_a = 0$ and $n_c = 1$ for 10 different values of σ_η and compute the mean and the variance of the error in the estimated model parameters, as shown in Figure 1. The algebraic algorithm without any iterative refinement estimates the parameters with an error[16] of less than 3.7% for the levels of noise considered. These errors are comparable to those of the Ferrari-Trecate and Bemporad's algorithms reported in [13] which are about $2 \sim 3\%$. The error is reduced significantly to about 1% (see Figure 1 left) by using the algebraic algorithm with iterative refinement via Expectation and Maximization (EM).

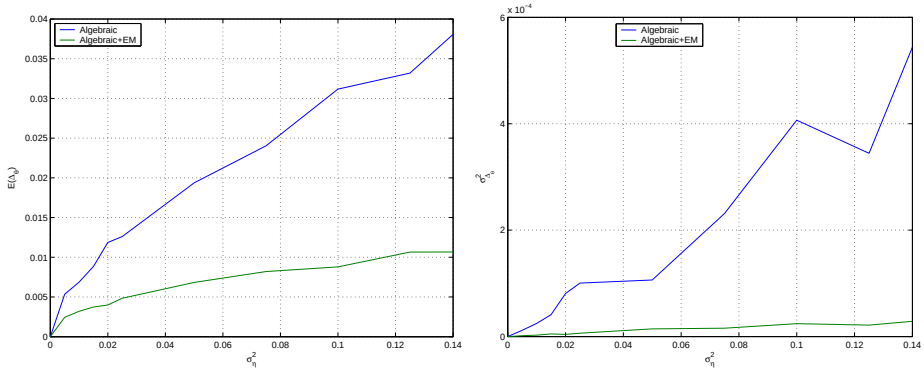

Fig. 1. Means (left) and variances (right) of the error in the estimation of the model parameters for different levels of noise. Blue curves are for the purely algebraic Algorithm 1; Green curves are for the EM algorithm initialized with the solutions from Algorithm 1.

Error as a Function of the Model Orders. Consider the PWAR system taken from [13]

$$y_t = \begin{cases} 2y_{t-1} + 0u_{t-1} + 10 + w_t & \text{if} \quad y_{t-1} \in [-10, 0], \\ -1.5y_{t-1} + 0u_{t-1} + 10 + w_t & \text{if} \quad y_{t-1} \in (0, 10], \end{cases} \tag{31}$$

with initial condition $y_0 = -10$, input $u_t \overset{\text{i.i.d.}}{\sim} \mathcal{U}(-10, 10)$ and noise $w_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 0.01)$.

We applied our algorithm[17] with known number of models $n = 2$, but unknown model orders (n_a, n_c). For all $\kappa > 1.3 \cdot 10^{-8}$, our algorithm correctly estimates the orders as

[16] The error between the estimated parameters $\hat{\boldsymbol{b}}$ and the true parameters $\boldsymbol{b}$ is computed as

$$\max_{i=1,\ldots,m.} \min_{j=1,\ldots,n.} \frac{\hat{\boldsymbol{b}}_i - \boldsymbol{b}_j}{[\mathbf{0}_{(K-1)\times 1}\ I_{K-1}]\boldsymbol{b}_j}.$$

[17] Since the ARX model is an affine model with a constant input, we slightly modify our algorithm by using homogeneous coordinates for the regressor $\boldsymbol{x}_t$, i.e., appending an entry of "1."

$n_a = 1$ and $n_c = 0$. For such orders, the estimates of the ARX model parameters are $[1.9878, 0, 10.0161]^T$ and $[-1.4810, 0, 10.0052]^T$, which have an error of 0.0020.

We also evaluated the performance of our algorithm as a function of the orders (n_a, n_c) for a known number of models $n = 2$. Rather than estimating the orders using formula (20), we use a fixed value for (n_a, n_c) and search for the polynomial in the null space of $L_n(n_a, n_c)$ with the smallest degree-lexicographic order. We repeat the experiment for multiple values of $n_a = 1, \ldots, 4$ and $n_c = 1, \ldots, 10$, to evaluate the effectiveness of equation (11) at finding the "correct" null space of $L_n(n_a, n_c)$. Figure 2 shows the results for $\kappa = 10^{-5}$. Notice that for all the range of values of n_a and n_c, the algorithm gives an error that is very close to the theoretical bound of 0.01 (the noise variance). These results are significantly better than those reported in [13] for the Ferrari-Trecate and Bemporad's algorithms when applied with wrong model orders. The results are comparable to those of Ferrari-Trecate and Bemporad's algorithms when applied with the true model orders.

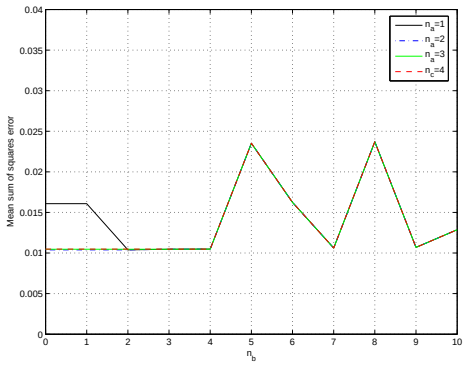

Fig. 2. Mean sum of squares error for various orders of the ARX models.

Experimental Results on Test Datasets We applied our algorithm with $n = n_a = n_c = 2$ to four datasets of $T = 60,000$ measurements from a component placement process in a pick-and-place machine [11]. For comparison with the results in [13], we first report results on a down-sampled dataset of 750 consisting of one out of every 80 samples. The 750 points are separated in two overlapping groups of points. The first 500 points are used for identification, and the last 500 points are used for validation. Table 1 shows the average sum of squared residuals (SSR) – one step ahead prediction errors, and the average sum of squared simulation errors (SSE) obtained by our method for all four datasets, as well as the SSE of Ferrari-Trecate's and Bemporad's algorithm for the first dataset as reported in [13]. It is worth mentioning that the SSE and SSR errors provided by our method are not strictly comparable to those [13]. This is because Ferrari-Trecate's and Bemporad's algorithms apply to PWARX models in which the mode λ_t is a piecewise linear function of the past inputs and outputs, while our method applies to switched ARX models in which λ_t can evolve arbitrarily. Therefore, for PWARX models λ_t is known automatically once the piece-wise linear map has been learned, while for switched ARX models one must use the measured output y_t to determine λ_t as in (28).

We also tested our algorithm on the entire datasets. We split the 60,000 measurements in two groups of 30,000 points each. The first 30,000 are used for identification and the last 30,000 for simulation. Table 2 shows the average sum of squared residual error (SSR) and the average sum of squared simulation error (SSE) obtained by our method for all four datasets. Figure 3 shows the true and simulated outputs for dataset 1.

Table 1. Training and simulation errors for down-sampled datasets.

Dataset	n n_a n_c	Our method's SSR	Our method's SSE	Ferrari-Trecate SSE	Bemporad SSE
1	2 2 2	0.0803	0.1195	1.98	2.15
2	2 2 2	0.4765	0.4678	N/A	N/A
3	2 2 2	0.6692	0.7368	N/A	N/A
4	2 2 2	3.1004	3.8430	N/A	N/A

Overall, the algorithm demonstrates a very good performance in all four datasets. The running time of a MATLAB implementation of our algorithm is 0.15 second for the 500 data points and 0.841 second for $30,000$ data points.

Table 2. Training and simulation errors for complete datasets.

Dataset	n n_a n_c	SSR	SSE
1 with all points	2 2 2	$4.9696 \cdot 10^{-6}$	$5.3426 \cdot 10^{-6}$
2 with all points	2 2 2	$9.2464 \cdot 10^{-6}$	$7.9081 \cdot 10^{-6}$
3 with all points	2 2 2	$2.3010 \cdot 10^{-5}$	$2.5290 \cdot 10^{-5}$
4 with all points	2 2 2	$7.5906 \cdot 10^{-6}$	$9.6362 \cdot 10^{-6}$

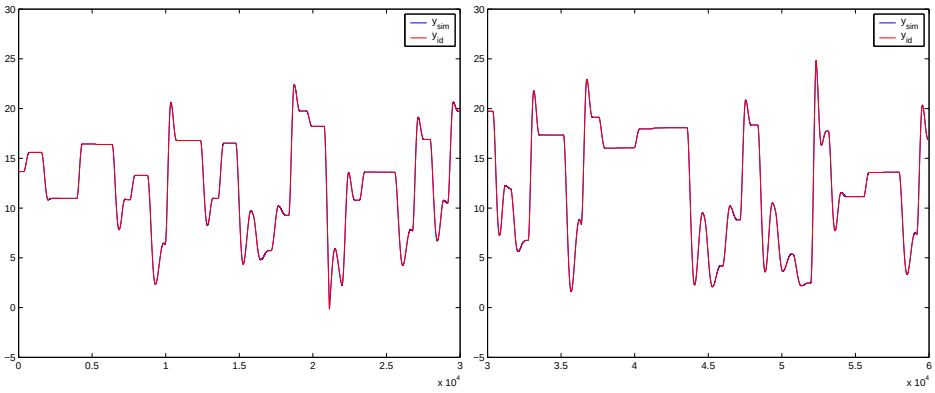

Fig. 3. Training and simulation sequences for complete datasets – the simulated and the identified sequences overlap almost exactly.

5 Conclusions

In this paper, we propose a linear-algebraic solution to the problem of identifying (deterministic) hybrid ARX systems. The algorithm can deal with the general case in which the switches are arbitrary and the number and orders of the constituent ARX systems are unknown. It can also tolerate moderate noises in the data. In the future, we would like to investigate efficient ways for on-line implementation of the algorithm as well as generalize our methods to state-space models.

References

1. A. Alessandri and P. Coletta. Design of Luenberger observers for a class of hybrid linear systems. In *Hybrid Systems: Computation and Control*, volume 2034 of *LNCS*, pages 7–18. Springer Verlag, 2001.

2. B.D.O. Anderson and C.R. Johnson. Exponential convergence of adaptive identification and control algorithms. *Automatica*, 18(1):1–13, 1982.
3. A. Balluchi, L. Benvenuti, M. Di Benedetto, and A. Sangiovanni-Vincentelli. Design of observers for hybrid systems. In *Hybrid Systems: Computation and Control*, volume 2289 of *LNCS*, pages 76–89. Springer Verlag, 2002.
4. A. Bemporad, G. Ferrari, and M. Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE Transactions on Automatic Control*, 45(10):1864–1876, October 2000.
5. A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino. A greedy approach to identification of piecewise affine models. In *Hybrid Systems: Computation and Control*, LNCS, pages 97–112. Springer Verlag, 2003.
6. A. Bemporad, J. Roll, and L. Ljung. Identification of hybrid systems via mixed-integer programming. In *IEEE Conference on Decision and Control*, pages 786–792, 2001.
7. A. Doucet, A. Logothetis, and V. Krishnamurthy. Stochastic sampling algorithms for state estimation of jump Markov linear systems. *IEEE Transactions on Automatic Control*, 45(1):188–202, 2000.
8. G. Ferrari-Trecate, D. Mignone, and M. Morari. Moving horizon estimation for hybrid systems. *IEEE Transactions on Automatic Control*, 47(10):1663–1676, 2002.
9. 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.
10. J. Harris. *Algebraic Geometry: A First Course*. Springer-Verlag, 1992.
11. A. Juloski, W. Heemels, and G. Ferrari-Trecate. Data-based hybrid modelling of the component placement process in pick-and-place machines. In *Control Engineeting Practice*. To appear.
12. K. Kanatani and C. Matsunaga. Estimating the number of independent motions for multibody motion segmentation. In *Asian Conference on Computer Vision*, 2002.
13. H. Niessen and A.Juloski. Comparison of three procedures for identification of hybrid systems. In *Conference on Control Applications*, 2004.
14. V. Pavlovic, J. M. Rehg, T. J. Cham, and K. P. Murphy. A dynamic Bayesian network approach to figure tracking using learned dynamic models. In *Proc. of the Intl. Conf. on Comp. Vision*, pages 94–101, 1999.
15. J. K. Tugnait. Detection and estimation for abruptly changing systems. *Automatica*, 18(5):607–615, 1982.
16. D. Del Vecchio and R. Murray. Observers for a class of hybrid systems on a lattice. In *Hybrid Systems: Computation and Control*, LNCS. Springer Verlag, 2004.
17. E.I. Verriest and B. De Moor. Multi-mode system identification. In *Internal Report 99-23, ESAT-SISTA, K.U.Leuven (Leuven, Belgium)*, 1999.
18. R. Vidal. *Generalized Principal Component Analysis (GPCA): an Algebraic Geometric Approach to Subspace Clustering and Motion Segmentation*. PhD thesis, University of California, Berkeley, August 2003.
19. R. Vidal. Identification of PWARX hybrid models with unknown and possibly different orders. In *IEEE Conference on Decision and Control*, 2004.
20. R. Vidal, A. Chiuso, and S. Soatto. Observability and identifiability of jump linear systems. In *Proc. of IEEE Conference on Decision and Control*, pages 3614–3619, 2002.
21. R. Vidal, A. Chiuso, S. Soatto, and S. Sastry. Observability of linear hybrid systems. In *Hybrid Systems: Computation and Control*, LNCS, pages 526–539. Springer Verlag, 2003.
22. 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 IEEE Conference on Decision and Control*, 2003.