43rd IEEE Conference on Decision and Control
December 14-17, 2004
Atlantis, Paradise Island, Bahamas

Identification of Hybrid Linear Time-Invariant Systems via Subspace Embedding and Segmentation (SES)

Kun Huang† Andrew Wagner‡ Yi Ma‡
†Department of Biomedical Informatics
Ohio State University
Columbus, OH 43210
khuang@bmi.osu.edu
‡Coordinated Science Laboratory
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign, IL 61801
{awagner,yima}@uiuc.edu

***Abstract*— This paper considers the offline identification of hybrid linear time-invariant (LTI) systems that are based on state-space models. This includes the identification of the number of LTI systems involved, the orders of the systems, and the switching times. By embedding the input/output data in a higher dimensional space, the problem of finding the switching times of the hybrid system becomes one of segmenting the data into distinct subspaces. Since these subspaces correspond to the original linear systems, their number and dimension must be found automatically. We examine and compare two different embedding methods. One is based on the well-known subspace method and the other is based on a direct input/output relationship. A robust and deterministic generalized principal component analysis (GPCA) algorithm is presented to solve the multiple-subspace identification problem. In addition, we show that data from near the switching points corresponds to points outside the subspaces under the embedding, and are thus readily identified by the GPCA algorithm. Although the resulting algorithm is purely algebraic, it is numerically robust and can tolerate moderate amounts of noise. Extensive simulations and experiments are presented to demonstrate the performance of the proposed algorithm and methods.**

***Index Terms*— hybrid system identification, subspace method, input/output embedding, subspace segmentation, generalized principal component analysis.**

I. INTRODUCTION

In this paper, we consider the identification problem of a class of hybrid linear time-invariant systems (LTI) based on state-space models. We assume that the state-space LTI models involved have the same number of inputs and outputs but probably different orders, [1] and

$$\text{Hybrid LTI:} \begin{cases} x_{t+1} &= A_{\lambda(t)}x_t + B_{\lambda(t)}u_t + v_t, \\ y_t &= C_{\lambda(t)}x_t + D_{\lambda(t)}u_t + w_t, \end{cases} \quad (1)$$

where the discrete switching state $\lambda(t)$ is a piecewise-constant function taking values in $\{1, 2, \ldots, s\}$, [2] $A_i \in \mathbb{R}^{n_i \times n_i}$, $B_i \in \mathbb{R}^{n_i \times l}$, $C_i \in \mathbb{R}^{m \times n_i}$, and $D_i \in \mathbb{R}^{m \times l}$ are the system matrices for the ith LTI system, $u_t \in \mathbb{R}^l$, $y_t \in \mathbb{R}^m$, and $x_t \in \mathbb{R}^{n_i}$ are the input, output, and state of the hybrid system, and v_t and w_t are the model and output noise, respectively. Regarding the above hybrid system, we are interested in the following problem:

Problem 1 (Hybrid ID): Given only the input u_t and the output y_t of system (1) over a period of time and an upper bound n on the order of all the LTI systems, determine:

1) the times where the discrete switching state changes value, i.e., the piecewise-constant function $\lambda(t)$
2) the number of discrete states s, i.e., the number of LTI systems involved in the hybrid system
3) the unknown order n_i of each LTI system
4) ultimately, all of the system parameters (A, B, C, D) and states $\{x_t\}$.

The reader should be aware that if the answers to the first 1)-3) sub-problems are known, especially the switching function $\lambda(t)$, then sub-problem 4) becomes a conventional identification problem for a single LTI system. Solutions to the single-system case have been well established in the literature [16], [25]. Therefore, in this paper, we focus on finding solutions to the first three sub-problems.

Relation to Previous Work. Filtering and identification of hybrid linear systems, especially jump-Markov linear systems, was an active area of research throughout the seventies and into the eighties; a review as of 1982 can be found in [24]. Recently, there has been a significant revival of interest in the observability and identifiability of hybrid systems, drawing attention simultaneously from control [1], [2], [3], [4], [10], [12], [15], signal processing [17], machine learning [8], [9], [10], [13], [18], and computer vision [20] researchers. There have also been a few approaches to systematically developing the theory of observability for discrete event and linear hybrid systems [5], [11], [19], [21], [22], [23], [26], [27]. For piecewise (input/output) ARX models, some effective algorithms for identifying the model parameters have already been proposed [6], [7], [12].

Because the number of LTI systems, the model parameters, the discrete state, and the switching function are *all unknown*, the identification problem is rather a challenging one in that there is a strong coupling between the estimation of the model parameters and the segmentation of (data into) discrete states. Traditionally, this is viewed as a “chicken-and-egg” problem, and the typical approach is to *alternate* between assigning data points to models and computing the model parameters from data points, starting from a

This work is partially supported by startup funding from Dept. of Biomedical Informatics in OSU and funding from DARPA and NSF.

[1] One can always view the states of all of the systems, independent of order, as being embedded in the state-space of their highest order. For the method presented in this paper, it does not make any difference if continuity in the states is imposed or not.

[2] We assume that the hybrid system is switching relatively slowly from one system to another. We will quantify how slow the switching needs to be in Section IV.

random or heuristic initialization [12], [6].[3] It was recently discovered that the above difficulty can be resolved *non-iteratively* for hybrid linear models via an algebraic procedure, known as *generalized principal component analysis* (GPCA) [28]. This method has been successfully applied to the identification of hybrid linear systems based on input/output ARX models [29].

Paper Contributions. This paper attempts to study identification algorithms for hybrid linear systems (1) based on state-space LTI models rather than input/output ARX models. We will show that, using the existing subspace method [25] or a method that we will introduce in this paper, we can embed the input/output data of each LTI system into a subspace, and therefore the main ideas of GPCA are still largely applicable to the state-space models. Using the embedding presented in this paper, a GPCA algorithm can simultaneously and automatically identify the number of systems, the orders of the state-space models, and the switching times all from the input/output data. The only requirements are that the inputs are random enough that the states visited span the observable subspace of each system, and that the switchings of the system are infrequent compared to the order of the system. The *recursive and robust* implementation of the GPCA presented in this paper is additionally capable of handling moderate amounts of noise in the measurements of the inputs and outputs of the system.

Paper Organization. In Section II, we discuss how to convert the hybrid identification problem into one that identifies multiple subspaces by embedding the input/output data into a higher dimensional space. In Section III, we show how to identify the multiple subspaces via GPCA. In Section IV, we show how to handle the embedded data points around the switching times since these points do not belong to any of the subspaces. In Section V, we demonstrate the proposed identification schemes through a series of experiments for both deterministic and stochastic systems.

II. Subspace Embeddings for an LTI System

In this section we study two different ways of embedding the input/output data from an LTI system as points of a subspace in a high-dimensional ambient space. The first embedding, based on the so-called oblique projection, was introduced as a part of the subspace method [25]; and the second embedding is based on the direct input/output relationship. Each embedding has its own desirable properties, as will be discussed in great detail.[4] Using either embedding, the hybrid identification problem becomes one of identifying multiple subspaces in the embedded data from all of the LTI systems. A solution to that problem will be given in Section III.

Notation. For a stochastic LTI system,

$$\text{LTI:} \begin{cases} x_{t+1} &= Ax_t + Bu_t + v_t, \\ y_t &= Cx_t + Du_t + w_t. \end{cases} \tag{2}$$

we adopt the notation of [25]. Let n be the order of the system, $x_t \in \mathbb{R}^n$ the state of the system, $u_t \in \mathbb{R}^l$ the input of the system, and $y_t \in \mathbb{R}^m$ the output of the system. Then we have $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times l}$, $C \in \mathbb{R}^{m\times n}$, and $D \in \mathbb{R}^{m\times l}$. Define the *ith-order observability matrix* as

$$\Gamma_i \doteq \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{i-1} \end{bmatrix} \in \mathbb{R}^{im\times n}.$$

For an observable system, $\text{rank}(\Gamma_n) = n$. Given a time instant t and two integers i, j, define the input and output block Hankel matrices, respectively, as

$$U_{t|t+i-1} \doteq \begin{bmatrix} u_t & \cdots & u_{t+j-1} \\ \vdots & \vdots & \vdots \\ u_{t+i-1} & \cdots & u_{t+i+j-2} \end{bmatrix} \in \mathbb{R}^{(li)\times j},$$

$$Y_{t|t+i-1} \doteq \begin{bmatrix} y_t & \cdots & y_{t+j-1} \\ \vdots & \vdots & \vdots \\ y_{t+i-1} & \cdots & y_{t+i+j-2} \end{bmatrix} \in \mathbb{R}^{(mi)\times j}.$$

Furthermore, define the input/output Hankel block matrix to be

$$W_{t|t+i-1} \doteq \begin{bmatrix} U_{t|t+i-1} \\ Y_{t|t+i-1} \end{bmatrix} \in \mathbb{R}^{(l+m)i\times j}. \tag{3}$$

For a pair of fixed i, j, define the *past* and *future* input Hankel block matrices as

$$U_p = U_{0|i-1} \text{ and } U_f = U_{i|2i-1},$$

and the *past* and *future* output Hankel block matrices as

$$Y_p = Y_{0|i-1} \text{ and } Y_f = Y_{i|2i-1}.$$

The Toeplitz matrix is denoted by

$$\Delta_i \doteq \begin{bmatrix} D & 0 & 0 & \cdots & 0 & 0 \\ CB & D & 0 & \cdots & \vdots & 0 \\ CAB & CB & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 & 0 \\ CA^{i-3}B & \vdots & \cdots & CB & D & 0 \\ CA^{i-2}B & CA^{i-3}B & \cdots & CAB & CB & D \end{bmatrix} \in \mathbb{R}^{(mi)\times(li)}.$$

A. Input/Output Embedding via the Oblique Projection

The subspace method introduced in [25] has proposed a way of embedding the input/output data of a system into a subspace. This scheme makes use of the notion of *oblique projection*. Consider three matrices A, B, and C with the same number of columns ($A \in \mathbb{R}^{p\times j}$, $B \in \mathbb{R}^{q\times j}$, $C \in \mathbb{R}^{r\times j}$). The *orthogonal projection* of the row space of A onto B is denoted by A/B and computed as

$$A/B = AB^T(BB^T)^\dagger B.$$

[3] This is similar to the expectation and maximization (EM) method in machine learning.

[4] Our studies pose an open question: What are *all* possible subspace embeddings and what different roles can these embeddings play in the hybrid ID problem?

The *oblique projection* of the row space of A onto the row space of C along the direction of the row space of B is denoted by $A/_B C$ and calculated as the formula:

$$A/_B C = A \begin{bmatrix} C^T & B^T \end{bmatrix} \left[\begin{pmatrix} CC^T & CB^T \\ BC^T & BB^T \end{pmatrix}^{\dagger} \right]_{\text{first } r \text{ columns}} C.$$

We summarize the results from the subspace method [25] as the following proposition:

Proposition 1 (Subspace Method): Consider a deterministic LTI system (i.e., $v_t = w_t = 0$), its input u and output y over a time interval $t = 0, 1, \ldots, T-1$, two integers i and j, and let $W_p = \begin{bmatrix} Y_p \\ U_p \end{bmatrix}$. If the following three conditions are satisfied:

1) *The covariance matrix of $U = \begin{bmatrix} U_p \\ U_f \end{bmatrix}$ has rank $2i$,*
2) *The row spaces of U_f and $X_p = [x_0, x_1, \ldots, x_{j-1}]$ have no nontrivial intersection,*
3) *rank $\begin{bmatrix} U_p \\ X_p \end{bmatrix}$ = rank $\begin{bmatrix} U_p - U_p/U_f \\ X_p - X_p/U_f \end{bmatrix}$,*

then we have

$$Z \doteq Y_f /_{U_f} W_p = \Gamma_i \left[x(i), \ldots, x(i+j-1) \right]. \quad (4)$$

For the stochastic case (with v_t, w_t being stationary processes), this statement hold asymptotically as $j \to \infty$.

That is, the oblique projection of Y_f on W_p along U_f, i.e., column vectors of the matrix Z, belongs to the range of the observability matrix Γ_i. Each condition above imposes some constraints on the possible choices of i and j. From the first condition, we know that $j \geq 2i$. From the second condition, we need $j > li + n$. Finally, from the third condition, we need $j - li \geq li + n$. For an observable system, Γ_i has rank n for $i \geq n$. However, in order for the above oblique projection always to span an n-dimensional *proper* subspace in the ambient space $\mathbb{R}^{mi}$, we need $i > n$ in case $m = 1$. So the lower bounds for i and j are

$$i \geq n + 1, \quad j \geq 2li + n \geq (2l+1)(n+1) - 1. \quad (5)$$

Note that the first two conditions above do not necessarily imply the third condition, as shown by the example below.

Example 1: Consider the system:

$$A = \begin{bmatrix} 0.603 & 0.603 & 0 & 0 \\ -0.603 & 0.603 & 0 & 0 \\ 0 & 0 & -0.603 & -0.603 \\ 0 & 0 & 0.603 & -0.603 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.1582 & 0.4724 & 0.7395 & -0.4528 \end{bmatrix}^T,$$

$$C = \begin{bmatrix} -0.5794 & 1.0751 & -0.5225 & 0.1830 \end{bmatrix}, \; D = \begin{bmatrix} -0.7139 \end{bmatrix}.$$

We have $n = 4$, $l = 1$, and $m = 1$. The system is observable since the observability matrix Γ_n has rank 4. We choose $i = n + 1 = 5$ and $j = (2l+1)(n+1) - 1 = (2 \times 1 + 1)(4+1) - 1 = 14$. Hence $Y_f \in \mathbb{R}^{5 \times 14}$. It can be easily verified that, with independent random inputs, the rank of $Y_f /_{U_f} W_p$ is 4, and equation (4) always holds. However, if we choose $j = 13$, the above condition 3) is not satisfied since the left side of the condition 3) has rank 9 while the right side has rank 8. Furthermore, the rank of $Y_f /_{U_f} W_p$ becomes 5, and equation (4) fails.

B. Direct Input/Output Embedding

The subspace obtained for each LTI system by the above method depends only on the model parameters (A, C) but not at all on (B, D). Hence it cannot differentiate between systems that differ only in (B, D), or at least not from the subspaces alone. In addition, the available sample points for such subspaces, i.e., columns of the matrix Z, depend on input/output data over a time window size of at least $T = 2i + j - 1 \geq (2l+3)(n+1) - 2$,[5] which can be very large. This is not very desirable in our context since each switch may corrupt a large number of sample points.

We will now present a more direct way of embedding the input/output data of an LTI system into a subspace structure. The dimensionality of this subspace structure is related to the rank of the observability matrix Γ. The key is that there is a natural rank condition associated with the input/output data matrix $W_{t|t+k} \in \mathbb{R}^{(l+m)(k+1) \times j}$ defined in equation (3).[6]

Theorem 1: For a deterministic LTI system and a given $k > 0$, assume the rank of the observability matrix Γ_{k+1} is $q \leq n$. We always have

$$\text{rank}\left(W_{t|t+k}\right) \leq l(k+1) + q. \quad (6)$$

For large enough j and $k \geq n$, the bound on the rank of $W_{t|t+k}$ is also tight if $U_{t|t+k}$ is full-rank and the system is observable.

Proof: Let $s \doteq l(k+1) + q + 1$. We only need to prove the case for $j \geq s$, which can be achieved by showing that the size of the maximal set of linearly independent column vectors of $W_{t|t+k}$ is smaller than s. Select any s column vectors from $W_{t|t+k}$ corresponding to time instants $t_1, \ldots, t_s$ and denote them as w_{t_i} for $i = 1, \ldots, s$. We want to show that the w_{t_i}'s are linearly dependent. Let g_p be the pth row of Γ_{k+1}. We can pick $p_1 < p_2 < \cdots < p_q$ such that $g_{p_1}, g_{p_2}, \ldots, g_{p_q}$ are a set of maximal linearly independent row vectors of Γ_{k+1}. Denote the stack of these q rows as $\bar{\Gamma}_{k+1}$. We can then define

$$\bar{w}_{t_i} = \left[u_{t_i}^T, \ldots, u_{t_i+k}^T, g_{p_1} x_{t_i}, \ldots, g_{p_q} x_{t_i} \right]^T \in \mathbb{R}^{l(k+1)+q},$$

for $i = 1, 2, \ldots, s$. Therefore, there exist $\alpha_1, \ldots, \alpha_s \in \mathbb{R}$ such that

$$\sum_{p=1}^{s} \alpha_p \bar{w}_{t_p} = 0, \quad (7)$$

which gives $\bar{\Gamma}_{k+1} \sum_{i=1}^{s} \alpha_i x_{t_i} = 0$ and $\sum_{p=1}^{s} \alpha_p u_{t_p+r} = 0, r = 0, \ldots, k$. This implies that $\sum_{p=1}^{s} \alpha_p y_{t_p+r} = C \sum_{p=1}^{s} \alpha_p x_{t_p+r} = CA^r \sum_{p=1}^{s} \alpha_p x_{t_p}$, for $r = 0, 1, \ldots, k$. Since $\bar{\Gamma}_{k+1}$ represents a maximal set of linearly dependent row vectors of Γ_{k+1}, we have $\Gamma_{k+1} \sum_{p=1}^{s} \alpha_p x_{t_p} = 0$ and consequently $\sum_{p=1}^{s} \alpha_p y_{t_p+r} = 0$ for $r = 0, \ldots, k$.

[5] Time window for W_p is from t to $t + i + j - 1$, and from $t + i - 1$ to $t + 2i + j - 2$ for Y_f and U_f.

[6] The reader should realize that the following theorem and its proof can be easily paraphrased in terms of properties of the (input/output) transfer function associated with an nth-order observable state-space model. However, our statement is more general since k can be larger than the system order. This is important when the actual system order is not known.

This means that the s columns of $W_{t|t+k}$ corresponding to $t_1, t_2, \ldots, t_s$ are also linearly dependent, which implies that the maximal set of linearly independent columns of $W_{t|t+k}$ contains no more than $l(k+1)+q$ columns and thus (6) holds. If the system is observable, for large enough j and $k \geq n$, the equality of (6) can be easily shown using transfer functions. The proof is omitted here due to space limitations. ■

Under the conditions of Theorem 1, with both l and k known, in principle, we can determine q, the dimension of the observability matrix Γ. This will be the system order n for an observable system if $k \geq n$.

Example 2: Consider the same system as in Example 1, where $n = 4$, $l = 1$, and $m = 1$. The system is observable since the observability matrix Γ_n has rank 4. We choose $k = 4$. Therefore, the dimension of the ambient space is $(l+m)(k+1) = 10$, $W_{t|t+k} \in \mathbb{R}^{10\times j}$, and $q = \text{rank}\,(\Gamma_k) = 4$. It can be verified that the rank of W is typically $l(k+1)+q = 9$ for independent input U and large j. If we replace the output matrix C with $[0, 0, -0.5225, 0.1830]$, then $q = \text{rank}\,(\Gamma_k) = 2$ and the rank of W becomes $l(k+1)+q = 1(4+1)+2 = 7$. For this case, if we further reduce k to be 3, we still have $q = 2$, but $W_{t|t+k} \in \mathbb{R}^{8\times j}$ and rank $\big(W_{t|t+k}\big) = l(k+1)+q = 1(3+1)+2 = 6$.

When $k \geq n$, we know that the range of W will always be a *proper* subspace of $\mathbb{R}^{(l+m)(k+1)}$ even if $m = 1$. How does this subspace depend on the system parameters (A, B, C, D)? For a general MIMO system, the answer is not entirely clear, except that we know the subspace does depend on all the system matrices, which is *different* from the embedding given in Section II-A. This can already be shown for any observable single-output (possibly multiple-input) LTI system:

Proposition 2: For an observable single-output deterministic LTI system and $k = n$, let $\Gamma_n^\dagger = \left(\Gamma_n^T\Gamma_n\right)^{-1}\Gamma_n^T$. The range of $W_{t|t+n}$ spans an $(l(n+1)+n)$-dimensional hyperplane in $\mathbb{R}^{(l+1)(n+1)}$ whose normal vector is given by the row vector of the following rank-1 matrix:

$$F_n \doteq \big[-A\Gamma_n^\dagger\Delta_n, 0_{n\times l}, A\Gamma_n^\dagger, 0_{n\times 1}\big] + \big[B, 0, \ldots, 0\big] \\ -\big[0_{n\times l}, -\Gamma_n^\dagger\Delta_n, 0_{n\times 1}, \Gamma_n^\dagger\big] \quad \in \mathbb{R}^{n\times(l+1)(n+1)}.$$

The proof uses the structure of the Toeplitz matrix Δ defined earlier and is fairly straightforward. We omit the details for brevity. In fact, the above statement is very intuitive: the row vector of F very much corresponds to the coefficients of the transfer function, $H(z)$, of the LTI system.

Note that the time window size on which the sample points, i.e., the column vectors of W, depend is $T = k \geq n$, which can be significantly *smaller* than $(2l+3)(n+1)-2$, the minimum of the previous embedding. However, the dimension of the ambient space is now $(l+m)(k+1) \geq (l+m)(n+1)$, which is typically *larger* than $m(n+1)$, the minimum dimension of the previous embedding, especially when the number of inputs, l, is large. We summarize the differences of the two embeddings in Table I.

TABLE I

COMPARISON OF THE MINIMUM DIMENSION AND TIME WINDOW SIZE OF THE TWO EMBEDDINGS OF SECTION II-A AND II-B

Embeddings	Ambient Space	Subspace	Time Window
Oblique	$m(n+1)$	n	$(2l+3)(n+1)-2$
Direct	$(m+l)(n+1)$	$(l+1)(n+1)-1$	n

III. IDENTIFYING MULTIPLE SUBSPACES VIA RECURSIVE GENERALIZED PCA (GPCA)

The previous section provides us with two ways of embedding the input/output data of an LTI system as a proper subspace of a certain ambient space. Thus, for a hybrid LTI system that switches among multiple LTI systems, using the upper bound of their orders, say n, in the above embeddings, the embedded input/output data will in general belong to *multiple* subspaces[7] of possibly different dimensions (depending on their actual orders) in the ambient space, except for the few data points around the switches.[8] Therefore, the problem of identifying multiple LTI systems from their input/output data becomes a problem of identifying multiple subspaces in an ambient space from sample points given on these subspaces. The latter is exactly the problem addressed by generalized principal component analysis (GPCA). We here give a brief and self-contained review of the main results without proof. The interested readers may refer to [28] and [14] for more technical details. Our review here will concentrate mostly on the case with an *unknown* number of subspaces (or, in our context, systems) of *unknown* dimensions (or, in our context, system orders).

We first summarize the key results of the GPCA method [28] for the case in which the number of subspaces (or LTI systems), say s, is known.

Theorem 2 (Algebraic GPCA): A collection of s subspaces $\cup_{i=1}^s S_i \subset \mathbb{R}^K$ can be described as the algebraic variety of an ideal generated by homogeneous polynomials of the form $p(\boldsymbol{x}) = \prod_{i=1}^s(\boldsymbol{c}_i^T\boldsymbol{x}) = \beta^T\nu_s(\boldsymbol{x}) = 0$, where $\boldsymbol{c}_i \in \mathbb{R}^K$ is a normal vector to the ith subspace S_i. When s is known and a sufficient number of points $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^N$ are given on the subspaces, one can estimate all such polynomials from the null-space of the embedded data matrix $L_s = [\nu_s(\boldsymbol{x}_1), \ldots, \nu_s(\boldsymbol{x}_N)]^T$, and the normal vectors $\{\boldsymbol{c}_i\}$ to the ith subspace from the derivative of the polynomials $\{p(\boldsymbol{x})\}$, $\{Dp(\boldsymbol{x})\}$, at any point $\boldsymbol{x} = \boldsymbol{x}_i$ in the ith subspace. The dimension of the ith subspace is obtained as $n_i = K - \text{rank}(\text{span}\{Dp(\boldsymbol{x}_i)\})$.

Let us now consider the case in which s is unknown. The simplest strategy is to start with $s = 1$ and increase s until there is at least one polynomial of degree s fitting all the data, i.e., until the matrix L_s drops rank. For such an s, we can use Theorem 2 to separate the data into s

[7] Be aware that Embedding A does not depend on (B, D), and Embedding B does not distinguish systems that have the same transfer function.

[8] We will discuss in more detail how to deal with such "outliers" around the switches in the next section.

subspaces. However, there is one problem with this simple strategy: when the subspaces have different dimensions, there might be polynomials of degree d strictly less than the correct s, i.e., $d < s$, that also fit all the points.[9] For instance, a polynomial that fits two arbitrary lines and one plane in $\mathbb{R}^3$ can be of degree 2 instead of degree 3. Therefore, for each of the s subspaces found so far, we need to repeat the above process again and, whenever possible, further separate the subspace into subspaces of even lower dimension.[10] In summary, we have the following Recursive-GPCA algorithm for points lying in an unknown number of subspaces of unknown dimensions:

---

function **Recursive-GPCA**($\boldsymbol{X}$)
$s = 1$;
repeat
  build a data matrix $L_s(\boldsymbol{X}) \doteq [\nu_s(\boldsymbol{x}_1), \ldots, \nu_s(\boldsymbol{x}_N)]^T$ via the Veronese map ν_s of degree s;
  if rank(L_s) drops rank **then**
    compute the basis $\{\beta_i\}$ of the right null space of L_s;
    obtain polynomials as $\{p_i(\boldsymbol{x}) \doteq \beta_i^T \nu_s(\boldsymbol{x})\}$;
    $\boldsymbol{Y} = \emptyset$;
    for $j = 1 : s$ **do**
      select a point $\boldsymbol{x}_j$ from $\boldsymbol{X} \setminus \boldsymbol{Y}$;
      obtain the subspace S_j spanned by the derivatives span$\{Dp_i(\boldsymbol{x}_j)\}$;
      find the subset $\boldsymbol{X}_j \subset \boldsymbol{X}$ that belong to the subspace S_j;
      $\boldsymbol{Y} \leftarrow \boldsymbol{Y} \cup \boldsymbol{X}_j$;
      Recursive-GPCA($\boldsymbol{X}_j$); (S_j now as the ambient space)
    end for
    $s \leftarrow s_{max}$;
  else
    $s \leftarrow s + 1$;
  end if
until $s \geq s_{max}$.

Combining the GPCA method for subspace identification with the subspace embeddings of the input/output data, the overall scheme for identifying a hybrid LTI system is outlined by the following diagram:

$$(U, Y) \xrightarrow[\text{oblique or direct}]{\text{Embedding}} \boldsymbol{X} = Z \text{ or } W \xrightarrow{\text{GPCA}} \cup_{i=1}^{s} S_i.$$

We now illustrate how the algorithm works with a numerical example.

Example 3 (A Numerical Example): Here we consider a hybrid LTI system consisting of three linear systems. The first system is a 1st-order system

$$A_1 = -0.85, \quad B_1 = 0.5, \quad C_1 = -0.6, \quad D_1 = 0.9.$$

The second system is a 3rd-order system

$$A_2 = \begin{bmatrix} 0.6 & 0.6 & 0 \\ -0.6 & 0.6 & 0 \\ 0 & 0 & 0.4 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0.4756 \\ -0.4202 \\ 0.7728 \end{bmatrix},$$
$$C_2 = \begin{bmatrix} 0.5801 & 0.3964 & 0.7186 \end{bmatrix}, \quad D_2 = \begin{bmatrix} 1 \end{bmatrix}.$$

The third system is the 4th-order system given in Example 1. The hybrid system is driven by random inputs u_t from $t = 1$ to 900. The switches among these linear systems are shown in Fig. 1 Top. The data points are generated by embedding the input/output data via the oblique projection introduced in Section II-A. In our implementation, we choose $i = 5$ and $j = 14$ for the embedding. For each time instant t, a data point z_t is chosen as the first column of $Z = Y_f/_{U_f} W_p$. If U, Y all come from the same system, z_t lies in the range of the observability matrix of the system. As shown in the second plot of Fig. 1, the GPCA algorithm first segments the data points into two 4-dimensional subspaces in the ambient space $\mathbb{R}^5$. The first group corresponds to the third system of order 4. The second group corresponds to the data points from both the first and second systems. This is expected since the data points of the first system (spanning a 1-dimensional subspace) and those of the second system (spanning a 3-dimensional subspace) together span a 4-dimensional subspace. In the next level of recursion, the first subspace remains unchanged; the second one is segmented into a 3-dimensional subspace for the second system and a 1-dimensional subspace for the first (Fig. 1 Bottom). These two new subspaces cannot be split further. Note that at both levels of recursion, points around the switching times are grouped into a special "outliers group 0" since those z_t's do not belong to any of the subspaces associated with the linear systems. We will discuss how such points are dealt

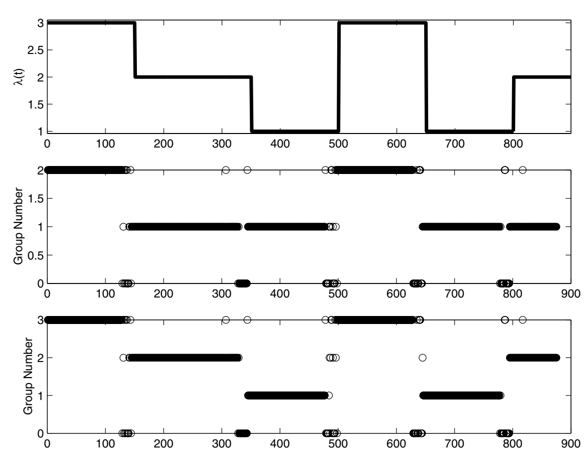

Fig. 1. Segmentation of input/output data for a hybrid LTI system consisting of three linear systems of order 1,3, and 4. Top: the ground-truth of the switching function $\lambda(t)$. Middle: the first level of recursion of GPCA on the input/output data embedded using the oblique projection. Bottom: the second level of recursion of GPCA – the segmentation is the same as the ground truth modulo a renumbering of the systems.

[9]That happens when the ideal generated by the sth-order homogeneous polynomials is not radical.

[10]One can rigorously show under mild algebraic conditions that if sufficient sample points are drawn from each subspace, then this recursive scheme is guaranteed to find the correct number of subspaces and their correct dimensions. Due to reasons of space, we omit the details of the proof.

with in more detail in the next section.[11]

In practice, real data sets will not be noise-free (i.e., $v_t, w_t \neq 0$). Hence, one must find a balance between the accuracy of the representation and the simplicity of the model. This can be achieved within a *robust* version of the above recursive GPCA algorithm by introducing an error tolerance τ when finding the points $\boldsymbol{X}_j$ belonging to each subspace. That is, the algorithm assigns a data point into the closest subspace only if the distance (or space angle) between the point and the subspace is within the error tolerance τ. For more details, we refer the reader to [14].

IV. Effect of Switches on Embedding and Segmentation of Subspaces

Note that for both of the embeddings mentioned above, each embedded data point depends on the system input and output from a time window, say T. If the window contains a switch from one system to another, the obtained data points, even in the noise-free case, may not belong to any of the subspaces associated with any individual system, as shown in Fig. 2. Since such points have very little geometric or algebraic structure, they should be treated as "outliers" for the purpose of hybrid system identification.

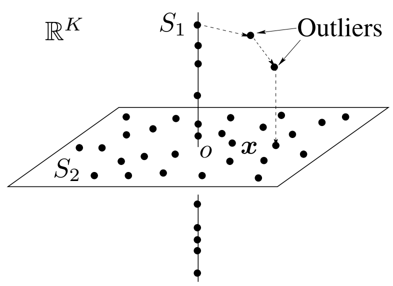

Fig. 2. Trajectory of the embedded points when switching from one subspace (system) to another shows how outliers are generated.

Therefore, in addition to the error tolerance τ, in the actual implementation of the algorithm, we should also tolerate a certain percentage, say δ, of outliers in the data points. Points that cannot be assigned to any subspaces within the given error tolerance τ are assigned to a special group of outliers. For a data set with N points, a valid solution must be one where the identified subspaces cover at least $(1-\delta)N$ points within the error tolerance τ. This is the reason why, in Fig. 1, such "transient" data points around the switches are segmented into an additional group 0. The ability to handle outliers is critical for the hybrid system identification problem.

The outlier tolerance ratio δ in effect sets realistic bounds on the number of switches allowed for the hybrid system during any given time period. From our analysis in Section II, if n is an upper bound on the order of all LTI systems involved, then the time window on which a data point depends is at least $k \geq n$ for the direct embedding and $2i+j-1 \geq (2l+3)(n+1)-2$ for the oblique embedding (see Table I). Therefore, given an outlier tolerance ratio δ, the number of switches allowed for the direct embedding is

$$N_s \leq \frac{\delta N}{k} \leq \frac{\delta N}{n}, \tag{8}$$

if N is the length of the overall time period. Similarly, the number of switches should be

$$N_s \leq \frac{\delta N}{2i+j-1} \leq \frac{\delta N}{(2l+3)(n+1)-2}, \tag{9}$$

for the oblique embedding.

A Tradeoff Between Precision and Dimensionality. Obviously, the direct embedding has the advantage of having a smaller time window, hence a lower percentage of outliers. This allows the subspaces, especially the switching times to be identified more precisely. However, it normally has a much higher data dimension – $(l+m)(n+1)$ compared to $m(n+1)$ in the minimum case for the embedding using oblique projection. Because the dimension of the Veronese map grows rapidly, the computational cost of the GPCA algorithm also grows rapidly with the dimension of the data. This paper does not advocate either embedding as *the* embedding of choice. Rather, we would like to point out via our studies that there seems to be a fundamental *tradeoff* between the time window size and the embedding dimension. The search for other embeddings with different characteristics is still under way.

V. Implementation and Experiments

In this section, we discuss implementation issues associated with the proposed algorithm, and present the results of our experimentation.

Setup of the robust GPCA algorithm. As mentioned in Section IV, the robust GPCA algorithm takes two parameters: the error tolerance τ and the outlier tolerance (ratio) δ. The tolerance for outliers is necessary for the detection of switching points, even for the case without noise. For a slowly switching system, we can set δ to be small, and equations (8) and (9) give a good estimate of the range of δ. In our experiments, we conservatively set δ to be 25%. The error tolerance τ is a parameter that the algorithm uses to decide whether or not to assign a data point to a subspace. If the space angle between a data point (as a vector) and the subspace is less than τ, the algorithm assigns the data point to the subspace. Therefore, the appropriate value of τ varies with the noise level. In our experiments, for a noise level from 0% to 2% (of the input and output), τ ranges from 0.02 to 0.1 for the embedding based on oblique projection; and τ is 0.025 for the direct embedding for the same noise range. All the algorithms are implemented in Matlab on a computer with 1.9 GHz CPU.

Post-processing. In the presence of both noise and outliers, the results of the GPCA algorithm are expected to have two types of errors. Firstly, some points can be assigned to a wrong subspace (and hence wrong system) if they

[11] Also notice that a very small number of points are mis-classified due to the fact that they are too close to the intersection of two subspaces.

are close to the intersection of two subspaces (e.g., there are a couple mis-classified points in Fig. 1). Secondly, sometimes (often because the error tolerance is set too small), the GPCA algorithm may over-split one subspace into multiple subspaces that do not correspond to actual systems. Therefore, some post-processing of the results is useful for repairing these errors.

For each subspace obtained, we can re-calculate its basis from the points using conventional principal component analysis (PCA). With this new basis, we reassign all the data points to the their closest subspaces.[12] To repair segmentation errors that cause the excessive splitting of subspaces, we compute the space angle between the obtained subspaces and merge any two subspaces that differ by a small space angle. The threshold for the space angle depends on the error tolerance τ. Typically, we set it to be 2τ.

Random inputs and slow switches. Our approach to the identification of the hybrid system depends on the segmentation of subspaces associated with the linear systems. Therefore, similar to other linear system identification methods, we require that the system is well excited, i.e., the input data is random enough so that the states visited span the entire observable subspace of each system. Furthermore, we require the switches between the systems to be slow enough due to the reason discussed in the previous section.

A. Subspace Segmentation and Detection of Switches

Fig. 3 shows segmentation resulting from the GPCA algorithm using two different embeddings of the input/output data of the hybrid system in Example 3. For this experiment the average norm of the input (u) and the output (y) is around 1. The input and output are corrupted by adding 0.5% independent Gaussian noise signals v_t and w_t. For the embedding using oblique projection, we choose $i = 5$ and $j = 14$,[13] and the embedded data points are in a 5-dimensional ambient space with three subspaces (i.e., , systems) of dimension 1, 3, and 4, respectively. For the direct embedding, we set $k = 4$. The embedded data points are in a 10-dimensional space with the subspaces of dimension 6, 8, and 9. As Fig. 3 shows, the robust GPCA algorithm along with post-processing successfully segments the data points into three groups (except for outliers in group number 0). The dimensions of the three subspaces are also correct. As expected, data points around the switches are classified as outliers, and the effect of switching is more significant in the embedding using oblique projection than that in the direct embedding. The mis-classification of data points can be further corrected by imposing the slow-switching assumption. In this experiment, the embedding using direct embedding has far fewer mis-classified data points but requires longer execution time (21 seconds compared to 6 seconds for embedding with oblique projection).

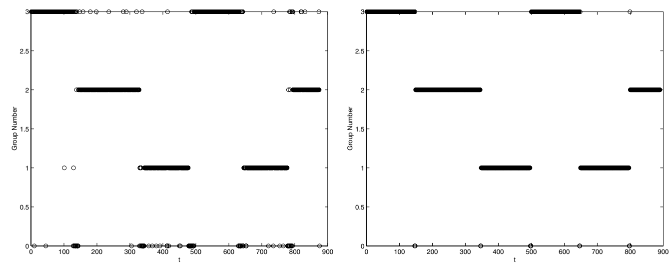

Fig. 3. Segmentation of data points for a hybrid system with 0.5% Gaussian noise added on both input and output ($\sigma = 0.05$). Left: input/output embedding using the oblique projection. Right: direct input/output embedding.

B. Subspace Angle Errors Induced by Noise

We also test the performance of the algorithm under different levels of Gaussian noise. We again use the hybrid system of Example 3. The subspaces associated with the individual linear systems are reasonably separated for both embeddings. We choose the noise level at $\sigma = 0$, 0.001, 0.002, 0.005, 0.01, and 0.02. For each noise level, we run the algorithm for 1000 trials. The three subspaces obtained are compared to the ground truth. The averaged errors (space angles between the subspaces and the true ones) are shown in Fig. 4. The direct embedding gives a smaller error than the embedding using oblique projection. Nevertheless, this is not an entirely fair comparison since the algorithm is dealing with subspaces of different dimensions in different ambient spaces. Notice that there are even angle errors at zero noise for the estimations. These errors are introduced by the points around the switching regions that are mis-classified.

C. Identifiability Using the Two Embeddings

The subspace obtained from the oblique projection is the range of the observability matrix of the linear system, which only depends on the matrices A and C. Therefore, this embedding is not able to distinguish between systems with different B and D. Conversely, the subspace obtained from the direct embedding depends on all matrices A, B,

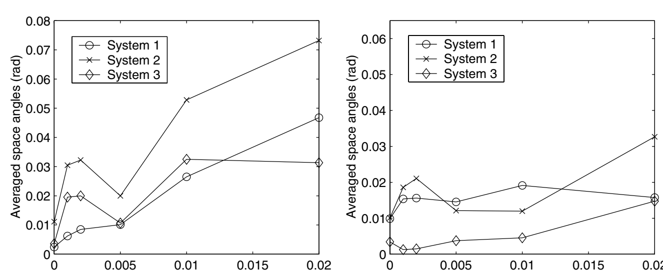

Fig. 4. Average subspace angle between the estimated subspaces and the true ones at different noise levels. Left: input/output embedding using the oblique projection. Right: direct input/output embedding.

[12]In principle, one can further "filter" the data points to reduce the mis-classified points by imposing the slow-switching assumption. Currently, no such filtering scheme is yet implemented in our experiments.

[13]This choice of i and j is the minimum in order to obtain a subspace structure for an order-4 system.

C, and D. To verify this experimentally, we choose a hybrid system that consists of three LTI systems. The first system is the same as the 4th-order system given in Example 3, the other two systems are variations of this system by modifying only B and D. As shown in Fig. 5, the GPCA algorithm cannot distinguish between the three systems with the embedding using oblique projection (Fig. 5 Left); yet with the direct embedding, the correct segmentation and dimensions of the subspaces are obtained (Fig. 5 Right). However, notice that the data points around the switches between the systems are correctly identified as outliers (group 0) for both embeddings.

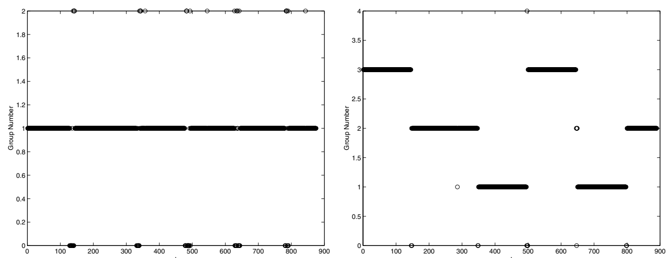

Fig. 5. Subspace embedding and segmentation for a hybrid system consisting of three 4th-order linear systems with the same matrices A and C but different matrices B and D. Left: input/output embedding using the oblique projection. Right: direct input/output embedding.

VI. CONCLUSIONS

In this paper, we have demonstrated that the identification of a slowly-switching hybrid LTI system can be converted to the GPCA problem (i.e., estimation and segmentation of multiple linear subspaces) by embedding the input/output data into a high-dimensional ambient space. We have implemented and compared two different types of embeddings. From our analysis and experimental verification, we see that the direct embedding in general gives better estimation and segmentation of the subspaces, but that it has a higher computational cost than the embedding using oblique projection. Although switches among systems result in points that do not lie on any of the subspaces, they can be segmented by the robust GPCA algorithm from the subspaces as a group of outliers, and the switches can thus be readily detected.

The methods proposed in this paper are for the most general case. In practice, if additional information about the hybrid system is known, such as the number and order(s) of the LTI systems involved, the method can be simplified and its performance further improved. We will investigate these special but important cases in future work.

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] M. Babaali, M. Egerstedt, and E. Kamen. An observer for linear systems with randomly-switching measurement equations. In *Proceedings of the 2003 American Control Conference*, pages 1879–1884, 2003.

[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] Y. Bar-Shalom and X.-R. Li. *Estimation and Tracking: Principles, Techniques, and Software*. Artech House, Boston MA, 1993.

[5] A. Bemporad, G. Ferrari, and M. Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE Trans. on Automatic Control*, 45(10):1864–1876, October 2000.

[6] 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.

[7] A. Bemporad, J. Roll, and L. Ljung. Identification of hybrid systems via mixed-integer programming. In *Proc. of IEEE Conf. on Decision and Control*, pages 786–792, 2001.

[8] M. Billio, A. Monfort, and C.P. Robert. Bayesian estimation of switching ARMA models. *Journal of Econometrics*, (93):229–255, 1999.

[9] A. Blake, B. North, and M. Isard. Learning multi-class dynamics. *Advances in Neural Information Processing Systems*, 11:389–395, 1999. MIT Press.

[10] A. Doucet, A. Logothetis, and V. Krishnamurthy. Stochastic sampling algorithms for state estimation of jump Markov linear systems. *IEEE Trans. on Automatic Control*, 45(1):188–202, 2000.

[11] J. Ezzine and A. H. Haddad. Controllability and observability of hybrid systems. *International Journal of Control*, 49(6):2045–2055, 1989.

[12] 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.

[13] Z. Ghahramani and G. E. Hinton. Variational learning for switching state-space models. *Neural Computation*, 12(4):963–996, 1998.

[14] K. Huang, Y. Ma, and R. Vidal. Minimum effective dimension for mixtures of subspaces: A robust GPCA algorithm and its applications. In *CVPR*, 2004.

[15] V. Krishnamurthy and J. Evans. Finite-dimensional filters for passive tracking of markov jump linear systems. *Automatica*, 34(6):765–770, 1998.

[16] L. Ljung. *System Identificaiton: theory for the user*. Prentice Hall, 1987.

[17] A. Logothetis and V. Krishnamurthy. Expectation maximization algorithms for MAP estimation of jump Markov linear systems. *IEEE Trans. on Signal Processing*, 47(8):2139–2156, 1999.

[18] K. Murphy. Switching Kalman filters. Technical report, U. C. Berkeley, 1998.

[19] C. Özveren and A. Willsky. Observability of discrete event dynamic systems. *IEEE Trans. on Automatic Control*, 35:797–806, 1990.

[20] V. Pavlovic, J. M. Rehg, and J. MacCormick. Learning switching linear models of human motion. In *NIPS*, 2000.

[21] P. Ramadge. Observability of discrete-event systems. In *Proc. of IEEE Conference on Decision and Control*, pages 1108–1112, 1986.

[22] A. Sun, S. S. Ge, and T. H. Lee. Controllability and reachability criteria for switched linear systems. *Automatica*, 38:775–786, 2002.

[23] F. Szigeti. A differential algebraic condition for controllability and observability of time varying linear systems. In *Proc. of IEEE Conference on Decision and Control*, pages 3088–3090, 1992.

[24] J. K. Tugnait. Detection and estimation for abruptly changing systems. *Automatica*, 18(5):607–615, 1982.

[25] P. van Overschee and B. De Moor. *Subspace Identification for Linear Systems*. Kluwer Academic Publishers, 1996.

[26] 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.

[27] 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.

[28] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (GPCA): Subspace clustering by polynomial factorization, differentiation, and division. Technical Report UCB/ERL, UC Berkeley, August 15 2003.

[29] 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.