On-line structured subspace identification with application to switched linear system

Laurent Bako, Guillaume Mercère, Stéphane Lecoeuche

On-line structured subspace identification with application to switched linear systems

Laurent Bako[1,2]*, Guillaume Mercère[3], Stéphane Lecoeuche[1,2]

1. Ecole des Mines de Douai,
Département Informatique et Automatique 59508 Douai, France

2. Laboratoire d'Automatique, Génie Informatique et Signal, UMR CNRS 8146
Université des Sciences et Technologies de Lille, 59655, France

3. Université de Poitiers
Laboratoire d'Automatique et d'Informatique Industrielle
40, Avenue du recteur Pineau, 86022 Poitiers, France

guillaume.mercere@univ-poitiers.fr
lecoeuche@ensm-douai.fr

Abstract

This paper is concerned with the identification of switched linear MIMO state-space systems in a recursive way. Firstly, a structured subspace identification scheme for linear systems is presented which turns out to have many attractive features. More precisely, it does not require any singular value decomposition (SVD) but is derived using orthogonal projection techniques; it allows a computationally appealing implementation and it is closely related to input-output models identification. Secondly, it is shown that this method can be implemented on-line to track both the range space of the extended observability matrix and its dimension and thereby, the system matrices. Thirdly, by making use of an on-line switching times detection strategy, this method is applied to blindly identify switched systems and to label the obtained submodels. Simulation results on noisy data illustrate the abilities and the benefits of the proposed approach.

Keywords: Subspace methods, multivariable systems identification, recursive methods, linear systems, switched systems, state space models.

*Corresponding author: bako@ensm-douai.fr

1 Introduction

Hybrid systems are models whose dynamical behavior switches among a certain number of submodels according to some discrete-event state. The identification of such systems from input-output data has been attracting a lot of research effort [1, 2, 3]. The main difficulty in inferring these models from data is related to the fact that the discrete state and the parameters are highly coupled and are both unavailable.

Most of the existing contributions on the subject deal with the class of Piecewise ARX models. These models are defined on a polyhedral partition of the regression space, each submodel being associated to one polyhedron. Hence, the main challenge in the identification of this class of systems is to determine the right partition of the regressors. Once this task is complete, the estimation of the different submodels follows by means of standard linear regression techniques [4]. To find the regions, it is proposed in [1] to group the regressors associated with each mode by performing the K-means clustering algorithm in a special space and then, to compute the parameters of each submodel. The solution in addition to requiring a knowledge of the system order is suboptimal. In a similar framework, [5] uses a statistical clustering approach instead and provides a method to derive the number of submodels from batch data, the orders being assumed to be available *a priori*. Another category of methods alternates between assigning the data to submodels and estimating simultaneously their parameters by using a weights learning technique [6], solving a Minimum Partition into Feasible Subsystems (MIN PFS) problem [7], or resorting to bayesian learning [8]. In [9], the hybrid system identification problem has been transformed into a linear or quadratic mixed integer programming problem for which there exist efficient tools for solving it in an optimal way. Conversely, this algorithm suffers from a high computational complexity. Another optimal, but deterministic algorithm is the algebraic geometric approach developed in [3]. Under the assumption that the data are perfect (in the sense that they are not corrupted by noise), the authors recast the problem into one of computing and deriving a homogeneous polynomial from which the submodels are deduced without any iteration [10]. For a comprehensive review of hybrid systems identification methods, we refer the interested reader to the survey paper [11].

So far, the methods mentioned apply to ARX models with in principle, arbitrary switches that is, there is no minimum dwell time required between mode changes. However, it can be noticed that the identification of these kinds of models (ARX with arbitrary switches) is mostly suboptimal and that optimality comes with some severe restrictions. By making the assumption of a minimum dwell time, state space models can be regarded as an alternative to these input-output models, and particularly in the case of multiple inputs multiple outputs (MIMO) systems. An argument in favor of the state space models is that many existing analysis methods (control, observer design, fault detection and isolation, ...) rely on them. When the switches are separated by a certain minimum time, subspace identification, coupled with detection techniques have been shown to be operative in batch mode. Among others, the references [12], [13], [14], [15] addressed recently this problem in an off-line context. Notice that in this case, as the state is generally unknown, there is no available regression vector so that partitioning the regression space (here the state-input space) becomes harder. The minimum dwell time is needed precisely to overcome this challenge.

A major problem of batch data methods for the identification of switched systems is that the collected data set may not cover all the operating modes. Therefore, all the modes that are not visited by the system during the collection time of this data, are inevitably ignored.

In this paper we propose a recursive subspace identification algorithm which realizes online the multiple tasks of estimation, detection and decision. To the best of our knowledge, there exist only two papers [16] and [17] that address the problem of recursive identification of switched systems. However, the algorithms developed apply to SISO ARX models. In the extension of our previous work [18], we focus here on state space models identification. Firstly, a structured subspace identification scheme is presented which differs from the standard approaches in that it does not require any singular values computation. The key point of our method is to keep control of the state basis in which the system matrices are to be computed. To this end, we worked out a special transformation which allows the handling of MIMO systems in a suitable canonical basis. Concretely, the state observability, initially distributed over all the outputs is encompassed into a single auxiliary output defined as a combination of all the outputs. An attractive feature of the method is that it can easily be extended for adaptive rank and subspace tracking. This enables us to estimate online the submodels of a switched linear MIMO system with possibly different orders. For this kind of system, each submodel may be slowly time-varying and from a submodel to another, the order may also change. For this reason, the order is adaptively identified along with the parameters.

The main focus of the paper is the identification of linear switched MIMO systems. Before coming properly to this point in Section 4, we briefly formulate in Section 2 the problem of identifying a single linear system. We also review briefly the principle of subspace methods and point out the need of elaborating a recursive method that would be able to provide the orders and the matrices in a constant basis. In Section 3, a structured subspace identification strategy, based on the conversion of the MIMO system into a single output system with the same state sequence, is developed. Section 4 presents the application of the new identification scheme to the on-line identification of switched systems and Section 5 provides simulation results to demonstrate the effectiveness of the scheme. Section 6 concludes the paper.

2 Identification of a single linear system

In this section and the following section, we present a subspace-based identification method that will be used in §4 for the recursive estimation of the constituent submodels of a linear switched system. To begin with, consider a linear system represented by the following discrete-time stochastic state space model

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) \\ \quad\;\; y(t) = Cx(t) + Du(t) + v(t), \end{cases} \tag{1}$$

where $u(t) \in \mathbb{R}^{n_u}$, $y(t) \in \mathbb{R}^{n_y}$, $x(t) \in \mathbb{R}^n$ and $v(t) \in \mathbb{R}^{n_y}$ are respectively the input, output, state and output-noise vectors. (A, B, C, D) are the system matrices relatively to a certain coordinate basis of the state space. It will be assumed that $\{x(t)\}$, $\{u(t)\}$, $\{y(t)\}$ and $\{v(t)\}$, $t \in \mathbb{Z}$, are all ergodic and (weakly) stationary stochastic processes. The model (1) is the so-called output-error model [19].
The considered identification problem can be stated as follows. Given realizations $\{u(t)\}_{t=1}^{N_s}$ and $\{y(t)\}_{t=1}^{N_s}$ of the input and output processes generated by a system of the form (1) on a finite but sufficiently wide time horizon, estimate the minimum dimension n of the state process and the matrices (A, B, C, D) up to a similarity transformation.
We start by making the following assumptions.

A1. The input process $\{u(t)\}$ is an ergodic and (weakly) stationary process that is Persistently Exciting of order at least $f + n$ [4], (more simply expressed by $\{u(t)\}$ *is PE*$(f + n)$) where $f > n$ will be defined later.

A2. The output noise $\{v(t)\}$ is a zero-mean white noise process and is statistically uncorrelated with the input $\{u(t)\}$. More explicitly, for all t, s, $\mathbf{E}[v(t)v(s)^\top] = \delta_{ts}\sigma_v^2 I_{n_y}$ and $\mathbf{E}[u(t)v(s)^\top] = 0$, where $\mathbf{E}[\,]$ denotes the expected value and δ is the Kronecker delta.

A3. The matrix A of the model (1) is asymptotically stable that is, all its eigenvalues are strictly inside the unit circle.

A4. The model (1) is minimal that is, (A, B) is reachable and (A, C) is observable.

To begin with the identification procedure, let $f > n$ be an integer and define for any time index t,

$$u_f(t) = \begin{bmatrix} u(t)^\top & \cdots & u(t+f-1)^\top \end{bmatrix}^\top \in \mathbb{R}^{f n_u}. \tag{2}$$

In a similar manner as $u_f(t)$, we also define the vectors $y_f(t)$ and $v_f(t)$. Then we formulate the data matrices $U_{1,f,N}$ and $X_{1,N}$ as

$$\begin{aligned} U_{1,f,N} &= \begin{bmatrix} u_f(1) & \cdots & u_f(N) \end{bmatrix}, \\ X_{1,N} &= \begin{bmatrix} x(1) & \cdots & x(N) \end{bmatrix}, \end{aligned} \tag{3}$$

where f and N are user-defined parameters obeying $n < f \ll N$. We let $Y_{1,f,N}$ and $V_{1,f,N}$ be defined similarly to $U_{1,f,N}$ in (3). From the system equations (1), one can, as it is customary in subspace identification [20], write the following embedded data equation

$$Y_{1,f,N} = \Gamma_f X_{1,N} + H_f U_{1,f,N} + V_{1,f,N}, \tag{4}$$

where Γ_f is the extended observability matrix and H_f the block Toeplitz matrix defined as

$$\Gamma_f = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{f-1} \end{bmatrix} \in \mathbb{R}^{fn_y \times n}, \quad H_f = \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ CA^{f-2}B & CA^{f-3}B & \dots & D \end{bmatrix} \in \mathbb{R}^{fn_y \times fn_u}.$$

Most of the subspace-based methods [19], [21] proceed by performing first orthogonal or oblique projection techniques on the data equation (4). Then, the Singular Value Decomposition (SVD) algorithm is used together with some rank conditions to retrieve both the system order and the range space of the extended observability matrix Γ_f in an *arbitrary* state space basis. The SVD is known to be numerically reliable but suffers conversely from a noticeable computational complexity and particularly in the context of recursive identification [22]. Hence, in order to apply the subspace concept in a recursive framework, it appears necessary to find some alternative algorithms to the SVD. Some reliable algorithms have been developed. For example, the IV-PAST method [23], borrowed from [24], has been introduced to track the observability subspace in a colored noise framework. However, as pointed out by the author, the state basis of the realization provided by that method may change during the estimation. The same remark holds also for the algorithm developed in [25]. But, it is desirable, regarding the case of switched systems, to get all the submodels in the same basis. Recently, the papers [26, 27] suggested an identification version of the Propagator Method [28] ordinarily used for subspace tracking in signal array processing. This latter scheme shares some strong similarities with the structured technique that will be presented in the next section.

3 Structured identification scheme for one linear model

As an alternative to the conventional subspace identification methods, we present in the following, a structured subspace identification method which does not require any SVD. The key idea in this method is to select in advance the state space basis of the model to be identified. In this way, not only the SVD step can be avoided but also the number of unknowns to be estimated is significantly reduced.

In order to present clearly the intuition behind our method we shall study first the case of MISO systems in §3.1 before coming to the case of MIMO systems in §3.2. We defer the estimation of the system matrices to §3.3 while the order is identified in §3.4. In the last part of this section (§3.5), we propose an extension of our algorithm to recursive identification.

3.1 MISO systems

It is well known that the state space matrices (A, B, C, D) are not uniquely determined since for any nonsingular matrix T, $(TAT^{-1}, TB, CT^{-1}, D)$ explains the input-output behaviour of the system in (1) as well. Suppose that the system in (1) is an observable MISO system. Then, the observability matrix $\Gamma_n = \Gamma_n(A, C)$ with $C \in \mathbb{R}^{1\times n}$, is a square and nonsingular real matrix. Using this matrix, a similarity transformation can be carried out by setting $x(t) \leftarrow \bar{x}(t) = \Gamma_n x(t)$. One can then easily show that the resulting dynamics matrix $A_c = \Gamma_n A \Gamma_n^{-1}$ is a companion matrix up to a transposition operation and that A and C have the forms

$$A_c = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & \cdots & -a_{n-1} \end{bmatrix}, \tag{5}$$

$$C_c = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}, \tag{6}$$

where the a_i are the coefficients of the characteristic polynomial p_A of A,

$$p_A(z) = \det(zI - A) = a_0 + a_1 z + \cdots + a_{n-1} z^{n-1} + z^n.$$

Throughout the paper, we will refer to (5)-(6) as the companion form of the state space representation.

3.2 MIMO systems

In the case of MIMO systems, the companion form (5)-(6) is no longer so straight to obtain. However, thanks to certain transformations to be detailed in the following, it is still possible to describe the system using the canonical companion form. We should however start by pointing out that A can be put in the form of A_c in (5) if and only if A is nonderogatory[1] [29]. It may occur that all the poles of the system are observable from a certain (single) output $y_j(t)$. This means that $\Gamma_n(A, c_j^\top)$, with $c_j^\top \in \mathbb{R}^{1\times n}$ the jth row of C, may have full rank. The state sequence may then be retrievable from only this output $y_j(t)$ similarly to the case of MISO systems. But this situation is far from always holding for a general MIMO system since the other outputs are likely to convey some dynamics which may not be visible from $y_j(t)$. One can find in the literature some attempts to directly estimate canonical state space representations based on Kronecker invariants, observability or controllability indices [30], [31]. A drawback of such schemes is that the Kronecker invariants are not always available (since we do not know the system matrices). The idea suggested in this paper allows to circumvent this inconvenience by recasting the system such that it can be handled as a single output one.

Consider an auxiliary output constructed as a linear combination of the system outputs defined as

$$y_a(t) = \sum_{j=1}^{n_y} \gamma_j y_j(t), \tag{7}$$

where the $y_j(t)$ are the components of the output vector $y(t)$ and the γ_j are real nonzero numbers. When A is nonderogatory, the γ_j's can be selected in such a way that the whole

[1] i.e., a matrix whose characteristic polynomial is equal to its minimal polynomial. Note that A is nonderogatory is a necessary condition for a MISO system to be observable.

dynamics of the system are observable from $y_a(t)$ only. Then, using only this blended output and the input measurements, it shall be possible to estimate A, B and linear combinations of the rows of C and D. Instead of doing so, let us replace just one component of the output vector (e.g. the first component) by $y_a(t)$. The vector $y(t)$ is hence changed into a vector $\bar{y}(t)$ that we define to be $y(t)$ with its first entry replaced by $y_a(t)$, i.e.,

$$\bar{y}(t) = K(\gamma)y(t), \tag{8}$$

with

$$K(\gamma) = \begin{bmatrix} \gamma_1 & \gamma_2 & \cdots & \gamma_{n_y} \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}. \tag{9}$$

Denote the new matrices by $\bar{C} = K(\gamma)C$, $\bar{D} = K(\gamma)D$ and $\bar{\Gamma}_f = \Gamma_f(A, \bar{C})$ and $\bar{H}_f = H_f(A, B, \bar{C}, \bar{D})$. Under this transformation, the data equation (4) can be rewritten as follows

$$\bar{Y}_{1,f,N} = \bar{\Gamma}_f X_{1,N} + \bar{H}_f U_{1,f,N} + \bar{V}_{1,f,N}, \tag{10}$$

where $\bar{Y}_{1,f,N}$ and $\bar{V}_{1,f,N}$ are the block Hankel matrices of the form (3) constructed respectively from $\bar{y}(t)$ and $\bar{v}(t) = K(\gamma)v(t)$.

Recall that the main expectation in this transformation is to reach an output $y_a(t) \in \mathbb{R}$ from which all the poles of the system (1) will be observable. This is equivalent to requiring $\Gamma_n(A, \bar{c}_1^\top)$, with $\bar{c}_1^\top = \gamma^\top C$, $\gamma = \begin{bmatrix} \gamma_1 & \cdots & \gamma_{n_y} \end{bmatrix}^\top$, to have full rank. If this is true, it will become possible to directly obtain, by setting a state transformation as $\bar{x}(t) \leftarrow \Gamma_n(A, \bar{c}_1^\top)x(t)$, the matrices of (1) in canonical form similarly as in the case of MISO systems. The question is, while $\Gamma(A, C)$ is unknown, how to find $\gamma_1, \cdots, \gamma_{n_y}$ so as to achieve this requirement. An answer to this question is given by Proposition 1. Before stating the proposition, we need some preliminary result.

Lemma 1. *Consider two matrices $A \in \mathbb{R}^{n\times n}$ and $C \in \mathbb{R}^{n_y \times n}$. Then, there exists $\gamma = \begin{bmatrix} \gamma_1 & \cdots & \gamma_{n_y} \end{bmatrix}^\top \in \mathbb{R}^{n_y}$ satisfying $\operatorname{rank}\left(\Gamma_n(A, \gamma^\top C)\right) = n$ if and only if A is nonderogatory and (A, C) is observable.*

Proof. Assume that there is $\gamma \in \mathbb{R}^{n_y}$ such that $\operatorname{rank}\left(\Gamma_n(A, \gamma^\top C)\right) = n$. Then

$$\Gamma_n(A, \gamma^\top C)A\Gamma_n(A, \gamma^\top C)^{-1}$$

is a matrix that has the companion form (5). Being similar to a companion matrix, we can conclude from a result in [29, p. 147] that A is nonderogatory. To see that (A, C) is observable, let us write

$$\Gamma_n(A, \gamma^\top C) = G\Gamma_n(A, C),$$

where

$$G = \begin{bmatrix} \gamma^\top & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \gamma^\top \end{bmatrix} \in \mathbb{R}^{n \times n n_y}.$$

With $\operatorname{rank}\big(\Gamma_n(A, \gamma^\top C)\big) = \operatorname{rank}\big(G\Gamma_n(A, C)\big) = n$, we have necessarily $\operatorname{rank}\big(\Gamma_n(A, C)\big) = n$ because otherwise, we would have by Sylvester's inequality, $\operatorname{rank}\big(G\Gamma_n(A, C)\big) \leq \operatorname{rank}\big(\Gamma_n(A, C)\big) < n$ for any $\gamma \in \mathbb{R}^{n_y}$. Therefore, (A, C) is observable.

Conversely, assume that A is nonderogatory and that (A, C) is observable. Then A is similar to a companion matrix of the form A_c defined in (5) [29]. Therefore, there is a nonsingular matrix L such that $\Gamma_n(A, C)L^{-1} = \Gamma_n(A_c, C_c)$, where $C_c = CL^{-1}$. Let ${c'}_j^\top$ be the jth row of C_c. Denote by $l_j(z)$ the polynomial of degree $n-1$ whose coefficients are defined by ${c'}_j^\top$ as $l_j(z) = {c'}_j(n)z^{n-1} + {c'}_j(n-1)z^{n-2} + \cdots + {c'}_j(1)$, where ${c'}_j(k)$ is the k-th entry of ${c'}_j$. Then, we know by Barnett's theorem [32, Theorem 1] that $\deg\big\{\gcd\big(l_1(z), \cdots, l_{n_y}(z), p_A(z)\big)\big\} = n - \operatorname{rank}\big(\Gamma_n(A_c, C_c)\big)$, where $p_A(z) = \det(zI_n - A_c)$. Since $\operatorname{rank}\big(\Gamma_n(A_c, C_c)\big) = n$, it follows that[2] $\gcd\big(l_1(z), \cdots, l_{n_y}(z), p_A(z)\big) = 1$. In other words, the polynomials $l_1(z), \cdots, l_{n_y}(z)$, $p_A(z)$ are relatively prime. This means that if we denote by $r_1, \cdots, r_n$ the roots of $p_A(z)$, then for any root r_j of $p_A(z)$, there exists $i \in \{1, \ldots, n_y\}$ such that $l_i(r_j) \neq 0$. Consequently, all the rows $\mathcal{L}_i^\top$ of the matrix

$$\mathcal{L} = \begin{bmatrix} \mathcal{L}_1^\top \\ \vdots \\ \mathcal{L}_n^\top \end{bmatrix} = \begin{bmatrix} l_1(r_1) & \cdots & l_{n_y}(r_1) \\ \vdots & \ldots & \vdots \\ l_1(r_n) & \cdots & l_{n_y}(r_n) \end{bmatrix} \in \mathbb{R}^{n \times n_y}$$

are nonzero. Here, $\mathcal{L}_i = \begin{bmatrix} l_1(r_i) & \cdots & l_{n_y}(r_i) \end{bmatrix}^\top \in \mathbb{R}^{n_y}$. Using again Barnett's theorem, one can see that $\operatorname{rank}\big(\Gamma_n(A, \gamma^\top C)\big) = \operatorname{rank}\big(\Gamma_n(A_c, \gamma^\top C_c)\big) = n$ is equivalent to saying that $\gamma_1 l_1(z) + \cdots + \gamma_{n_y} l_{n_y}(z)$ and $p_A(z)$ have no common root, that is, $\mathcal{L}_i^\top \gamma \neq 0$, $i = 1, \ldots, n$. The existence of such a γ follows from the fact that all the rows of $\mathcal{L}$ are nonzero.
If this were not the case, then for any γ, there would be an $i \in \{1, \ldots, n\}$ such that $\mathcal{L}_i^\top \gamma = 0$. But this would mean that $\mathbb{R}^{n_y} \subset \mathscr{A} \triangleq (\mathcal{L}_1)^\perp \cup \cdots \cup (\mathcal{L}_n)^\perp$, where $(\mathcal{L}_i)^\perp$ is the linear hyperplane that is orthogonal to the vector $\mathcal{L}_i$ and the symbol $\cup$ refers to sets union operation. On the other hand, since $\mathscr{A}$ is a subset of $\mathbb{R}^{n_y}$, we would have $\mathbb{R}^{n_y} = \mathscr{A}$. With $\mathcal{L}_i \neq 0$ for all i, this is impossible. Therefore, we can conclude that there is at least one $\gamma \in \mathbb{R}^{n_y}$ such that $\operatorname{rank}\big(\Gamma_n(A, \gamma^\top C)\big) = n$. □

Proposition 1. *Assume that the pair (A, C) of the system in (1) is observable and that the matrix A is nonderogatory. Let $\gamma = \begin{bmatrix} \gamma_1 & \cdots & \gamma_{n_y} \end{bmatrix}^\top \in \mathbb{R}^{n_y}$ be a vector generated randomly from a uniform distribution. Then, it holds with probability one that*

$$\operatorname{rank}\big(\Gamma_n(A, \gamma^\top C)\big) = n,$$

where $\Gamma_n\big(A, \gamma^\top C\big)$ is the observability matrix related to the blended output $y_a(t)$.

[2] gcd refers here to the greatest common divisor, taken here to be a monic polynomial.

Proof. Let $P(\gamma) = \det\left(\Gamma_n\left(A, \gamma^\top C\right)\right)$ be the determinant of $\Gamma_n\left(A, \gamma^\top C\right) \in \mathbb{R}^{n\times n}$. Then P is a polynomial with respect to the entries of γ. Consider the set $\mathcal{S} = \{\gamma \in \mathbb{R}^{n_y}/P(\gamma) = 0\}$, of all γ such that $\text{rank}(\Gamma_n(A, \gamma^\top C)) < n$. Consider the uniform probability measure denoted by P_r, to be defined on a σ-algebra $\mathcal{R}$ (that includes $\mathcal{S}$) over $\mathbb{R}^{n_y}$. Since (A, C) is observable and A is nonderogatory, we know from Lemma 1 that there is at least one γ^* such that $\text{rank}\left(\Gamma_n(A, {\gamma^*}^\top C)\right) = n$ and so, $P(\gamma^*) \neq 0$; which implies that the polynomial P is not identically null. Then, $\mathcal{S}$ is a proper algebraic set in the probability space $(\mathbb{R}^{n_y}, \mathcal{R}, \mathrm{P}_r)$ that is of dimension (Hausdorff dimension with respect to the euclidean metric) strictly less than n_y. From the measure theory [33], a subset such as $\mathcal{S}$ is known to be a null set. Thus, the complement of $\mathcal{S}$ is of full measure. In other words, the property $\text{rank}\left(\Gamma_n(A, \gamma^\top C)\right) = n$ holds almost surely, that is, with probability one. □

From the previous analysis it follows that either for MISO or for MIMO systems, a certain canonical companion state space representation can be reached directly from the identification process. To this purpose, we exhibited in each case a nonsingular matrix T which can be used to change the state coordinates basis in the equation (4).

Let $T = \Gamma_n(A, \bar{c}_1^\top)$ and $\mathcal{I}_i = \begin{bmatrix} e_i & e_{i+n_y} & \cdots & e_{i+(f-1)n_y} \end{bmatrix} \in \mathbb{R}^{fn_y\times f}$, where e_j is the vector in $\mathbb{R}^{fn_y}$ which has 1 in its jth entry and 0 anywhere else. We define a permutation matrix $S \in \mathbb{R}^{fn_y\times fn_y}$ as

$$S = \begin{bmatrix} \mathcal{I}_1 & \cdots & \mathcal{I}_{n_y} \end{bmatrix}^\top \in \mathbb{R}^{fn_y\times fn_y}. \tag{11}$$

Then, by multiplying $\bar{\Gamma}_f$ on the left by S, we obtain the following partition

$$S\bar{\Gamma}_f = \left[\begin{array}{c} \Gamma_n(A, \bar{c}_1^\top) \\ \hline \Gamma_{f-n}(A, \bar{c}_1^\top)A^n \\ \Gamma_f(A, \bar{c}_2^\top) \\ \vdots \\ \Gamma_f(A, \bar{c}_{n_y}^\top) \end{array}\right] = \left[\begin{array}{c} I_n \\ \hline \mathscr{P} \end{array}\right] T, \tag{12}$$

where the lower part of $S\bar{\Gamma}_f$ has been written as a linear combination $\mathscr{P}T$, with $\mathscr{P} \in \mathbb{R}^{(fn_y-n)\times n}$, of the rows of the submatrix T. This is possible because $\text{rank}\left(S\bar{\Gamma}_f\right) = \text{rank}\,(T) = n$. Therefore, the identification of $\bar{\Gamma}_f$ requires only the estimation of $\mathscr{P}$. It turns out that this matrix corresponds to the so-called Propagator defined and utilized in signal array processing [28] and recently in recursive subspace identification of MISO systems [26]. From now onwards, we may refer to $\mathscr{P}$ as the propagator. Now we are about to set up the basis in which we would like the system matrices to be retrieved. Recall that our approach makes it possible to choose this basis.

Realization in a basis determined by S. Given the transformation (8) of system (1), there are actually different ways to set up the permutation matrix S defined in (11). To obtain an equation of the form (12), the only requirement to be fulfilled is that the first n rows of $S\bar{\Gamma}_f$ must be linearly independent. The matrix T will then consist of these n linearly independent rows. Therefore, the choice of S determines a certain basis of the state. To estimate the range space of the extended observability matrix in such a basis, one can proceed as follows. From (12), note that the observability matrix is as [3]

$$\bar{\Gamma}_f = S^\top \left(S\bar{\Gamma}_f\right) = S^\top \begin{bmatrix} I_n \\ \mathscr{P} \end{bmatrix} T. \tag{13}$$

By applying a state transformation $x(t) \leftarrow Tx(t)$, we get $A \leftarrow TAT^{-1}$ and $\bar{C} \leftarrow \bar{C}T^{-1}$ so that $\bar{\Gamma}_f \leftarrow \bar{\Gamma}_f T^{-1}$. This means that T can be dropped from Eq. (13). Therefore, one can immediately extract the matrices A and $\bar{C}$ once $\mathscr{P}$ is known. This may be done as traditionally, by exploiting the A-shift invariance property of $\bar{\Gamma}_f$:

$$A = (\bar{\Gamma}_f^\uparrow)^\dagger \bar{\Gamma}_f^\downarrow \quad \text{and} \quad \bar{C} = \bar{\Gamma}_f(1:n_y,:), \tag{14}$$

with $\bar{\Gamma}_f^\uparrow = \bar{\Gamma}_f(1:(f-1)n_y,:)$, $\bar{\Gamma}_f^\downarrow = \bar{\Gamma}_f(n_y+1:fn_y,:)$; the symbol † refers to the generalized Moore-Penrose inverse.

Realization in companion form. In order to obtain a A-matrix in the companion form, we consider the partitioning matrix S to be defined as in (11). Starting from Eq. (12), if we let $\Omega = \begin{bmatrix} I_n & \mathscr{P}^\top \end{bmatrix}^\top$, then we have

$$\Gamma_f(A, \bar{c}_1^\top) = \Omega\,(1:f,:)\,T, \tag{15}$$

where $\Omega(1:f,:)$ refers to the first f rows of Ω. A similar transformation $x(t) \leftarrow Tx(t)$ induces $A \leftarrow TAT^{-1}$, $\bar{C} \leftarrow \bar{C}T^{-1}$ and hence $\Gamma_f(A,\bar{c}_1^\top) \leftarrow \Gamma_f(A,\bar{c}_1^\top)T^{-1}$. Therefore, we can drop the matrix T from (15) so that it remains $\Gamma_f(A,\bar{c}_1^\top) = \Omega\,(1:f,:)$ in the new basis. Note now that $\Omega\,(2:n+1,:) = \Omega\,(1:n,:)\,A = A$. Therefore, one can obtain A and $\bar{c}_1^\top$ in the companion form (5)-(6) without computing any Moore-Penrose inverse, that is:

$$A = \Omega\,(2:n+1,:) \quad \text{and} \quad \bar{c}_1^\top = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{1\times n}. \tag{16}$$

Similarly, by considering the global matrix $S\Gamma_f(A,\bar{C})$, we also get $\bar{c}_j^\top = \Omega\,((j-1)f+1,:)$, $j = 2,\cdots,n_y$. Finally, the C-matrix of the initial system (1) can be obtained as $C = K(\gamma)^{-1}\bar{C}$.

Thus, an additional attractive feature of the proposed structured identification scheme is that, contrarily to most of the subspace identification methods, it makes it possible to avoid computing the pseudo-inverse of $\bar{\Gamma}_f^\uparrow$ as in (14). This is quite comfortable in a recursive identification context, since the computation of the pseudo-inverse relies generally on the expensive SVD.

Once the matrices A and C are available in a certain basis, B and D can be estimated by a linear regression, under the assumption that the system is asymptotically stable (see e.g [22] for more details).

In the remainder of the paper we will adopt whenever possible, simplified notations as

[3] Recall that, as a permutation matrix, S satisfies $S^\top S = SS^\top = I$

$\bar{Y} = \bar{Y}_{t,f,N}$, $\bar{\Gamma} = \bar{\Gamma}_f$ and so forth. Therefore, Eq. (10) can be re-written simply as

$$\bar{Y} = \bar{\Gamma} X + \bar{H} U + \bar{V}, \tag{17}$$

where $\bar{V}$ has been constructed from $\bar{v}(t) = K(\gamma) v(t)$.

3.3 Estimation of extended observability matrix

In this subsection, we consider the problem of estimating the matrix $\mathscr{P}$. Inspired by the Multivariable Output Error State Space (MOESP) class of subspace methods [19], the first step in the estimation of $\mathscr{P}$ consists in eliminating the term $\bar{H}U$ in the data equation (17), by projecting the whole equation onto the orthogonal complement of the row space of U. In this objective, let us follow the RQ implementation method [19]. This results in the following proposition which is indeed similar to the one derived in [34] but with different assumptions.

Proposition 2. *Let the assumptions A1-A4 hold and the A-matrix be nonderogatory. Let the partitioning matrix S be set as in (11) and the RQ factorization of the input-output data matrix be given as*

$$\begin{bmatrix} U \\ \bar{Y} \end{bmatrix} = \begin{bmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}, \tag{18}$$

where $\bar{Y}$ is defined from $\bar{y}(t) = K(\gamma) y(t)$ by (3), and $\gamma \in \mathbb{R}^{n_y}$ is such that $\gamma^\top \gamma = 1$ and $T = \Gamma_n(A, \gamma^\top C)$ is nonsingular. Then,

$$\lim_{N \to \infty} \left(\frac{1}{N} S R_{22} R_{22}^\top S^\top \right) = \begin{bmatrix} \Sigma_{\mathsf{z}} & \Sigma_{\mathsf{z}} \mathscr{P}^\top \\ \mathscr{P} \Sigma_{\mathsf{z}} & \mathscr{P} \Sigma_{\mathsf{z}} \mathscr{P}^\top \end{bmatrix} + \sigma_v^2 \bar{R}_v, \tag{19}$$

where $\mathscr{P} \in \mathbb{R}^{(f n_y - n) \times n}$ is the matrix that appears in (12),

$$\Sigma_{\mathsf{z}} = \lim_{N \to \infty} \left(\frac{1}{N} \mathsf{Z} \Pi_U^\perp \mathsf{Z}^\top \right) \in \mathbb{R}^{n \times n}, \tag{20}$$

$$\Pi_U^\perp = I_N - U^\top (U U^\top)^{-1} U, \tag{21}$$

$\mathsf{Z} = TX$*, and*

$$\bar{R}_v = \begin{bmatrix} I_f & \gamma_2 I_f & \cdots & \gamma_{n_y} I_f \\ \gamma_2 I_f & I_f & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{n_y} I_f & 0 & \cdots & I_f \end{bmatrix}. \tag{22}$$

Proof. The proof is reported in Appendix B. □

Now, by considering Eq. (19), we introduce the notation

$$\Sigma = \left[\begin{array}{c|c} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{array}\right] = \bar{\Sigma} + \sigma_v^2 \bar{R}_v, \tag{23}$$

where

$$\bar{\Sigma} = \left[\begin{array}{c|c} \Sigma_{\mathsf{z}} & \Sigma_{\mathsf{z}}\mathscr{P}^\top \\ \mathscr{P}\Sigma_{\mathsf{z}} & \mathscr{P}\Sigma_{\mathsf{z}}\mathscr{P}^\top \end{array}\right].$$

Then, $\mathscr{P}$ can be estimated by minimizing the following cost function

$$\mathcal{J}(\mathscr{P}) = \|\Sigma_{21} - \mathscr{P}\Sigma_{11}\|_F^2, \tag{24}$$

whose (suboptimal) solution is given by

$$\mathscr{P} = \Sigma_{21}\Sigma_{11}^{-1}. \tag{25}$$

It should be noticed that the estimate obtained here for $\mathscr{P}$ may be biased in the presence of noise. One way to address this problem may be for example, to use the instrumental variable method [22]. However, regarding the application in Section 4 to the identification of switched systems, this may not be efficient since the effect of such a noise decorrelation technique is mainly asymptotic.

Remark 1.

- *It can be seen from Eq. (25) that computing $\mathscr{P}$ does not require the computation of the entire matrix Σ from (23). The knowledge of its first f (or n if it is known) rows or columns will be enough.*

- *From (16), one can also notice that not all the matrix $\mathscr{P}$ is exploited in the determination of A and C. Only some n_y rows of $\mathscr{P}$ need actually to be known.*

- *From these two observations, it appears that A and C can be extracted, after eliminating some redundancies, from the following data matrices:*

$$\bar{\bar{Y}} = \begin{bmatrix} y_{a,f}(t) & y_{a,f}(t+1) & \cdots & y_{a,f}(t+N-1) \\ y_{2:n_y}(t) & y_{2:n_y}(t+1) & \cdots & y_{2:n_y}(t+N-1) \end{bmatrix},$$
$$\bar{\bar{U}} = \begin{bmatrix} u_f(t) & u_f(t+1) & \cdots & u_f(t+N-1) \end{bmatrix},$$

where $y_{a,f}(t) = \begin{bmatrix} y_a(t) & y_a(t+1) & \cdots & y_a(t+f-1) \end{bmatrix}^\top$, $y_a(t) \in \mathbb{R}$ is the blended output defined in (7) and $y_{2:n_y}(t) = \begin{bmatrix} y_2(t) & \cdots & y_{n_y}(t) \end{bmatrix}^\top$ is the 2nd through the n_yth entries of the output vector at time t. This is a remarkable reduction in the dimension of the data matrices since contrarily to (3) (fn_y rows) $\bar{\bar{Y}}$ contains only $f + n_y - 1$ rows.

3.4 Estimation of the order

In order to conveniently estimate the matrix $\mathscr{P}$ by (25), we need to identify the order n of the system (1). Under some mild assumptions, a procedure for the estimation of the order is suggested in this subsection. Generally, from the point of view of subspace identification schemes, the order results almost always from an analysis of the singular values of the matrix R_{22} in (18) for example. Here, we would like to characterize the order without resorting to the expensive SVD so that an on-line application becomes possible. To this purpose, we assume that a strict upper bound $r_{\max} = f > n$ of the order is known. Then, the idea is to exploit the interesting structure of the matrix Σ defined in Eq. (23). An estimation of the order in a deterministic framework is first discussed before coming to the more challenging stochastic case which is treated in the last paragraph of this subsection.

3.4.1 Dealing with noise-free data

To proceed, we consider recursively a submatrix of Σ in (23) of the form $\Delta_r = \Sigma(1:r, 1:r)$, r running from $r_{\min}$ towards $r_{\max}$ with $r_{\min} < n < r_{\max}$. In view of (20), we require that $\operatorname{rank}(\lim_{N\to\infty} \frac{1}{N} X\Pi_U^{\perp}) = n$ which is equivalent, under the assumption that the system (1) is minimal, to requiring the input process $\{u(t)\}$ to be PE$(f+n)$ [35] (see also Proposition 4 in Appendix A). This assumption implies that the covariance matrix $\Sigma_{\mathbf{z}}$ defined in (20) is positive definite. Consequently, any square submatrix of the form $\Delta_r = \Sigma_{\mathbf{z}}(1:r, 1:r)$ is also positive definite. In the light of these precisions, Δ_r is nonsingular as long as $r \leq n$ but becomes singular as soon as $r > n$. Therefore, the order must satisfy $n = \max\{r : \operatorname{rank}(\Delta_r) = r\}$. These arguments justify the rank tracking algorithm described below.

Consider $\Delta_r = \bar{\Sigma}(1:r, 1:r)$ and let us write

$$\Delta_{r+1} = \bar{\Sigma}(1:r+1, 1:r+1) = \begin{bmatrix} \Delta_r & w_{r+1} \\ w_{r+1}^{\top} & s_{r+1} \end{bmatrix},$$

with $w_{r+1} = \bar{\Sigma}(:, r+1)$ and $s_{r+1} = \bar{\Sigma}(r+1, r+1)$. By the matrix inversion lemma [36], if Δ_r is nonsingular, then Δ_{r+1} is also nonsingular if and only if

$$h_{r+1} = s_{r+1} - w_{r+1}^{\top} \Delta_r^{-1} w_{r+1} \tag{26}$$

is nonzero. The reason is that, when Δ_{r+1}^{-1} exists, it can be obtained by

$$\Delta_{r+1}^{-1} = h_{r+1}^{-1} \begin{bmatrix} h_{r+1}\Delta_r^{-1} + \varphi_{r+1}\varphi_{r+1}^{\top} & \varphi_{r+1} \\ \varphi_{r+1}^{\top} & 1 \end{bmatrix}, \tag{27}$$

where $\varphi_{r+1} = -\Delta_r^{-1} w_{r+1} \in \mathbb{R}^r$. Hence, to identify the order, a possibility may be to compute recursively the inverse of Δ_r extracted from Σ going from $r = r_{\min}$ towards $r = r_{\max}$ until the order is detected. If we assume $\Delta_{r_{\min}}^{-1}$, with $r_{\min} \geq 1$, to be known, a starting value for r can be taken as $r = r_{\min}$. We then proceed to the computation of h_{r+1}; if $h_{r+1} = 0$ for some r^*, then Δ_{r^*+1} is singular and the conclusion $n = r^*$ is drawn; conversely, if $h_{r+1} \neq 0$, the recursion is pursued by computing Δ_{r+1}^{-1}, and then h_{r+2} and so on. The procedure is stopped when it becomes evident that Δ_{r+1} is singular, i.e., when $h_{r+1} = 0$. To see why, notice from (23) that for $r = n$, we have $w_{n+1} = \Delta_n \mathscr{P}(1, :)^{\top}$ and

$s_{n+1} = \mathscr{P}(1,:)\Delta_n \mathscr{P}(1,:)^\top$. Hence, by the definition (26) of h_{r+1}, we have

$$\begin{aligned} h_{n+1} &= s_{n+1} - w_{n+1}^\top \Delta_n^{-1} w_{n+1} \\ &= \mathscr{P}(1,:)\Delta_n \mathscr{P}(1,:)^\top - \mathscr{P}(1,:)\Delta_n \Delta_n^{-1} \Delta_n \mathscr{P}(1,:)^\top = 0. \end{aligned}$$

At the end of the loop, one can obtain $\Sigma_{11}^{-1} = \Sigma_{\mathsf{z}}^{-1} = \Delta_n^{-1}$ and so, $\mathscr{P}$ may be computed as in Eq. (25).

3.4.2 Dealing with noisy data

In the presence of noise, σ_v^2 in Eq. (23) is no longer null. Therefore, h_{n+1} will probably be greater than zero but our method can still be efficiently performed using a convenient threshold comparison. Naturally, this threshold will depend on the level of the noise that is acting on the process. It needs also to be related to the system we wish to identify and hence, has to be computed or adapted somehow, particularly in the case of switched systems (see §4).

The presence of noise tends to increase all the quantities h_r but a gap is still observable in their values when the iteration process reaches the rank of Σ unless the noise is dominant compared to the signal. Note that, owing to the assumption that Σ_{z} is positive definite and the Schur complement theorem [36] (the parameter h_{r+1} is indeed the Schur complement of Δ_r in Δ_{r+1}), we know that h_r is a positive scalar for $r \leq n$. It follows from (26) that

$$h_{r+1} = s_{r+1} + \sigma_v^2 - w_{r+1}^\top \Delta_r^{-1} \left(I_r + \sigma_v^2 \Delta_r^{-1}\right)^{-1} w_{r+1}.$$

If we consider that all the eigenvalues of Δ_r are significantly greater than the noise variance, then the spectral radius of $\sigma_v^2 \Delta_r^{-1}$ is lower than one. Therefore, by expanding the term in brackets at the first order we get the following approximation

$$h_{r+1} \approx \left(s_{r+1} - w_{r+1}^\top \Delta_r^{-1} w_{r+1}\right) + \sigma_v^2 \left(1 + w_{r+1}^\top \Delta_r^{-2} w_{r+1}\right),$$

which is composed of the signal part and the noise contribution. As in the previous subsection, when $r = n$, the first term vanishes so that $h_{n+1} \approx \sigma_v^2 \left(1 + \varphi_{n+1}^\top \varphi_{n+1}\right)$, where φ_r is defined as in (27). Therefore, a threshold can be chosen as

$$\text{Thres}(r) = T_0 \left(1 + \varphi_{r+1}^\top \varphi_{r+1}\right), \tag{28}$$

where T_0 denotes a constant, supposed to be slightly greater than the noise variance σ_v^2. Note that when the order is detected, we can approximate *a posteriori* the variance of the noise as:

$$\hat{\sigma}_v^2 \approx \frac{h_{n+1}}{1 + \varphi_{n+1}^\top \varphi_{n+1}}.$$

3.5 Recursive identification algorithm

In the previous parts, a complete off-line identification scheme has been investigated. From now on, we are interested in working out an on-line version of this procedure in order (as we will see in Section 4) to apply it to the estimation of switched systems. The on-line version of our algorithm relies indeed on the recursive adaptation of the matrix Σ defined in (23) using the new data available. At each time instant, Σ is updated first and then, the procedure described above for identifying both the order and the extended observability matrix is turned on.

Assume that an RQ factorization of the input-output data matrix in (18) is known at the instant $t = N + f - 1$. We would like then to update the R-part of this factorization when a new column is added to the concatenated input-output data matrix $\begin{bmatrix} U^\top & \bar{Y}^\top \end{bmatrix}^\top$. At time $t+1$, this matrix can be written as follows

$$\begin{bmatrix} \sqrt{\lambda} \begin{bmatrix} R_{11}(t) & 0 \\ R_{21}(t) & R_{22}(t) \end{bmatrix} & \begin{matrix} u_f(\bar{t}+1) \\ \bar{y}_f(\bar{t}+1) \end{matrix} \end{bmatrix} \begin{bmatrix} Q_1(t) & 0 \\ Q_2(t) & 0 \\ 0 & 1 \end{bmatrix},$$

where $\lambda < 1$ is a forgetting factor, $\bar{t} = t - f + 1$. To bring back the triangular form of the R-part, [22] suggested the use of an appropriate sequence of Givens rotations [37] gathered in a matrix $\mathbf{G}_1(t+1)$. By applying this method, we have the following.

$$\begin{aligned} &\begin{bmatrix} \sqrt{\lambda} \begin{bmatrix} R_{11}(t) & 0 \\ R_{21}(t) & R_{22}(t) \end{bmatrix} & \begin{matrix} u_f(\bar{t}+1) \\ \bar{y}_f(\bar{t}+1) \end{matrix} \end{bmatrix} \mathbf{G}_1(t+1) \\ &= \begin{bmatrix} R_{11}(t+1) & 0 & 0 \\ R_{21}(t+1) & \sqrt{\lambda} R_{22}(t) & z_f(t+1) \end{bmatrix}. \end{aligned} \tag{29}$$

It then turns out that

$$R_{22}(t+1) = \begin{bmatrix} \sqrt{\lambda} R_{22}(t) & z_f(t+1) \end{bmatrix},$$

and hence

$$S R_{22}(t+1) R_{22}^\top(t+1) S^\top = \lambda S R_{22}(t) R_{22}^\top(t) S^\top + S z_f(t+1) z_f^\top(t+1) S^\top. \tag{30}$$

From (19), it appears that the matrix Σ defined in (23) is a sort of covariance matrix. In the recursive framework with an exponentially decreasing width of the data window, we may compute $\Sigma(t+1)$ by dividing the quantity

$$S R_{22}(t+1) R_{22}^\top(t+1) S^\top = \sum_{k=0}^{t+1} \lambda^{t+1-k} S z_f(k) z_f^\top(k) S^\top$$

by the sum of the weights, namely $\alpha(t+1) = 1 + \lambda + \cdots + \lambda^{t+1} = \frac{1-\lambda^{t+2}}{1-\lambda}$. However, we are interested here in just extracting the matrix $\mathscr{P}$ from the particular structure of the matrix Σ defined in (23). Therefore, since there is no risk of divergence with $\lambda < 1$, we will just set

$$\Sigma(t+1) = S R_{22}(t+1) R_{22}^\top(t+1) S^\top$$

so that Eq. (30) becomes

$$\Sigma(t+1) = \lambda\Sigma(t) + Sz_f(t+1)z_f^\top(t+1)S^\top. \tag{31}$$

As already mentioned, it is not required that the matrix $\Sigma(t)$ is entirely adapted. If the order might vary, then it is essential to adapt all the first $r_{\max}$ columns or rows. After that, the procedure described above for the identification of the order, can be run. Two options are then possible: either a lower bound $r_{\min}$ of the order is known and then $\Delta_{r_{\min}}^{-1}$ is recursively adapted (using the matrix inversion lemma [36]) together with $\Sigma(t)$, or this information is not available and in this case, the procedure is started from $r = 1$. The whole algorithm for adaptive subspace and rank tracking is summarized in Algorithm 1.

Algorithm 1 On-line subspace tracking algorithm

```
• Initialization: set $\lambda$, $T_0$, $f$ and initialize $\Sigma(0)$, $\Delta_{r_{\min}}^{-1}(0)$.
• For $t = 1, \dots, \infty$
    1. Update the RQ factorization of the data matrix as in Subsection 3.5 and obtain $z_f(t)$.
    2. Update $\Sigma(t)$ using (31).
    3. Update $\Delta_{r_{\min}}^{-1}(t)$ using for example the matrix inversion lemma.
    4. Compute the order:
        – Set $r \leftarrow r_{\min}$
          While $h_{r+1} \geq \mathrm{Thres}(r)$ and $r < r_{\max}$
          Compute $\Delta_{r+1}^{-1}(t)$ by the formula (27);
          $r \leftarrow r + 1$;
          EndWhile
        – $n \leftarrow r$;
    5. Once the order is known, compute $\mathscr{P}(t)$ using the formula (25).
    6. Deduce the system matrices as in subsection 3.3.
• EndFor
```

4 Application to the identification of Switched Linear Systems (SLS)

In Sections 2 and 3, we have studied the problem of identifying a single linear model from input-output data in both batch and recursive modes. A new structured subspace identification method has been introduced for the estimation of the orders and the parameters of linear MIMO systems. By requiring fewer parameters to be estimated and providing the system matrices in a constant and known basis, this method turns out to be appropriate for a recursive processing of the data.

In this section, we shall extend this method to the identification of Switched Linear MIMO Systems (SLS). The considered switched system is described by the following state space model.

$$\begin{cases} x(t+1) = A_{\mu_{t+1},\mu_t} x(t) + B_{\mu_{t+1},\mu_t} u(t) \\ \quad\; y(t) = C_{\mu_t} x(t) + D_{\mu_t} u(t) + v(t), \end{cases} \tag{32}$$

where $u(t) \in \mathbb{R}^{n_u}$, $y(t) \in \mathbb{R}^{n_y}$ and $x(t) \in \mathbb{R}^{n_{\mu_t}}$ are respectively the input, output and state vectors. The subscript $\mu_t \in \{1, 2, \ldots\}$ refers to the discrete state which is assumed to be an unknown deterministic sequence; n_{μ_t} is the dimension of the state process at time t, $A_{\mu_{t+1},\mu_t} \in \mathbb{R}^{n_{\mu_{t+1}} \times n_{\mu_t}}$, $B_{\mu_{t+1},\mu_t} \in \mathbb{R}^{n_{\mu_{t+1}} \times n_u}$, $C_{\mu_t} \in \mathbb{R}^{n_y \times n_{\mu_t}}$, $D_{\mu_t} \in \mathbb{R}^{n_y \times n_u}$ are the system matrices at time t and $\{v(t)\} \in \mathbb{R}^{n_y}$ stands for a zero-mean white noise process. As in the case of the linear system (1), the stochastic processes $\{u(t)\}$, $\{x(t)\}$, $\{y(t)\}$ and $\{v(t)\}$ indexed by the set $\mathbb{Z}$ of integers, are assumed to be ergodic and weakly stationary. We also assume that Assumptions A1-A4 hold for each individual linear subsystem of (32).

The model (32) can loosely be understood as a generalized switched linear system by analogy with the definition of Generalized Jump Markov Linear Systems (GJMLS) [38]. Here, the occurrences of two consecutive switches are assumed to be reasonably separated so that rectangular matrices of the form A_{μ_{t+1},μ_t} show up rarely. Hence, the A-matrices are mostly square except at the switching times.

We denote the matrices A_{μ_{t+1},μ_t} and B_{μ_{t+1},μ_t} respectively by A_{μ_t} and B_{μ_t} in the case where the states $x(t+1)$ and $x(t)$ are of the same dimension. In this way, for $\mu_t = j$, we can more simply use (A_j, B_j, C_j, D_j) and n_j to refer to the matrices and the order of the jth submodel. But when $x(t+1)$ and $x(t)$ have different dimensions we may assume the transition matrices A_{μ_{t+1},μ_t} to be for example of the form

$$\begin{aligned} A_{\mu_{t+1},\mu_t} &= \mathcal{T}_{\mu_{t+1},\mu_t} A_{\mu_t} \\ B_{\mu_{t+1},\mu_t} &= \mathcal{T}_{\mu_{t+1},\mu_t} B_{\mu_t}, \end{aligned} \tag{33}$$

with

$$\mathcal{T}_{\mu_{t+1},\mu_t} = \begin{cases} \begin{bmatrix} I_{n_{\mu_{t+1}}} & 0 \end{bmatrix}, \text{ if } n_{\mu_{t+1}} < n_{\mu_t}, \\ \begin{bmatrix} I_{n_{\mu_t}} \\ 0 \end{bmatrix}, \text{ if } n_{\mu_{t+1}} > n_{\mu_t}, \\ I_{n_{\mu_t}}, \text{ if } n_{\mu_{t+1}} = n_{\mu_t}. \end{cases} \tag{34}$$

Given observations $\{u(t)\}_{t=1}^{\infty}$ and $\{y(t)\}_{t=1}^{\infty}$ of the input and output processes generated by a model such as (32), we are interested in recursively estimating the parameters $\big(A_j, B_j, C_j, D_j\big)$, the number of submodels as well as their orders $\{n_j\}_{j=1,2,\ldots}$. In order to achieve properly this task, we make the assumption that whenever the system visits a discrete state μ_t, it stays in it during a certain minimum time that we shall throughout the paper, refer to as the dwell time τ_{dwell}. On the other hand, the method to be presented does not require the number of submodels to be finite nor the orders $\{n_j\}$ to be equal. Furthermore, no constraint other than that of a minimum dwell is imposed on the switching mechanism.

With the assumption that the system stays long enough in each discrete state, we propose to apply online the identification method developed above for gradually learning the parameters of the submodels as they appear. In so doing, the number of submodels need not be finite or known. An additional motivation is that, we shall be able to track possible parameter variation of the constituent submodels of the system. Also, note that in practice, the minimum dwell time that is expressed in number of samples, is not that restrictive because too fast commutations may actually result in severe problems of stability.

We propose to apply Algorithm 1 to the identification of the SLS (32). As we will see, the switching times are detected based on the order estimation algorithm and the estimated submodels are recorded and labelled using simple classification techniques. We are first interested in analyzing what happens, when a switch occurs, in the system equations in terms of the order provided by Algorithm 1.
To proceed, we introduce the notations $\Gamma_q^i = \Gamma_q(A_i, C_i)$ and $H_q^i = H_q(A_i, C_i, B_i, D_i)$. Similarly, we let $\Psi_f^i = \Gamma_f\big(A_i, \bar{c}_i^\top\big) \in \mathbb{R}^{f\times n_i}$, $\mathcal{H}_f^i = H_f\big(A_i, B_i, \bar{c}_i^\top, \bar{d}_i^\top\big) \in \mathbb{R}^{f\times f n_u}$, where $\bar{c}_i^\top = \gamma^\top C_i$, $\bar{d}_i^\top = \gamma^\top D_i$ and γ is as above, a weight vector that satisfies rank $\big(\Gamma_n(A_i, \gamma^\top C_i\big) = n_i$, $i = 1, 2, \ldots,$. Let $u_f(t)$ and $y_f(t)$ be defined as in (2). For easy reference we also define

$$\Omega_{n_i}^i = \begin{bmatrix} A_i^{n_i-1}B_i & \cdots & A_iB_i & B_i \end{bmatrix}$$

$$\mathcal{R}^i = \begin{bmatrix} a_0^i & \cdots & \cdots & a_{n_i-1}^i \\ 0 & a_0^i & \cdots & a_{n_i-2}^i \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_0^i \end{bmatrix} \quad \text{and} \quad \mathcal{S}^i = \begin{bmatrix} 1 & & & \\ a_{n_i-1}^i & 1 & \text{\Large 0} & \\ \vdots & \vdots & \ddots & \\ a_1^i & \cdots & \cdots & 1 \end{bmatrix}, \tag{35}$$

where $a_0^i, \cdots, a_{n_i-1}^i$ are the coefficients of the characteristic polynomial of A_i. Finally, from the matrices in (35), we define

$$\mathcal{M}^{ij} = \left[\Big(\mathcal{R}^i\Psi_{n_i}^i + \mathcal{S}^i\Psi_{n_i}^j T_{j,i} A_i^{n_i}\Big) \quad \mathcal{S}^i\Big[\ \Big(\Psi_{n_i}^j T_{j,i} - \Psi_{n_i}^i\Big)\Omega_{n_i}^i \quad \Big(\mathcal{H}_{n_i}^j - \mathcal{H}_{n_i}^i\Big) \ \Big] \right], \tag{36}$$

where we recall that $T_{j,i}$ is the transition matrix (34) from submodel i to submodel j.

Before the switch. Note that the embedded data equation of the form (10) still holds as long as all the data involved are generated by the same linear submodel. In order to exploit this fact, consider that only the submodel i has been active on $[\tau - \tau_{\text{dwell}}, \tau - 1]$, where τ is a switching time and τ_{dwell} is the dwell time. Let the noise process $\{v(t)\}$ in (32) be identically null. Since we are interested here in studying the order estimated by Algorithm 1, we will just focus on the MISO model whose input is $u(t)$ and output is the blended output $y_a(t)$. Define $\nu = \tau - \tau_{\text{dwell}}$, an integer f that satisfies $\max(n_j) < f \ll \tau_{\text{dwell}}$, and

$$\mathcal{X}_{\nu|\bar{t}} = \begin{bmatrix} x(\nu) & \cdots & x(\bar{t}) \end{bmatrix} \in \mathbb{R}^{n_i\times(\bar{t}-\nu+1)}, \tag{37}$$

$$\mathcal{U}_{\nu|\bar{t},f} = \begin{bmatrix} u_f(\nu) & \cdots & u_f(\bar{t}) \end{bmatrix} \in \mathbb{R}^{fn_u\times(\bar{t}-\nu+1)}, \tag{38}$$

$$\mathcal{Y}_{\nu|\bar{t},f} = \begin{bmatrix} y_{a,f}(\nu) & \cdots & y_{a,f}(\bar{t}) \end{bmatrix} \in \mathbb{R}^{f\times(\bar{t}-\nu+1)}, \tag{39}$$

where $\bar{t} = t - f + 1$, t is the current time instant, $y_a(t)$ is the blended output defined in (7) and $u_f(k)$ and $y_{a,f}(k)$, $k = \nu, \cdots, \bar{t}$, are defined as in (2) from respectively $u(t)$ and $y_a(t)$.

Let now the following assumption of persistency of excitation hold.
Assumption A5: The input signal $\{u(t)\}$ is persistently exciting of order at least $f > n_i$ (for the submodel i) in the sense that there exists $\bar{t} \in [\nu, \tau - f]$ satisfying[4]

$$\operatorname{rank}\left(\begin{bmatrix} \mathcal{X}_{\nu|\bar{t}} \\ \mathcal{U}_{\nu|\bar{t},f} \end{bmatrix}\right) = fn_u + n_i,$$

From the data matrices (37)-(39), we can write the equation

$$\mathcal{Y}_{\nu|\bar{t},f} = \Psi_f^i \mathcal{X}_{\nu|\bar{t}} + \mathcal{H}_f^i \mathcal{U}_{\nu|\bar{t},f}, \tag{40}$$

where $\bar{t} = t - f + 1$, and $\bar{t}$ is assumed to be such that Assumption A5 is fulfilled. By using the Cayley-Hamilton's theorem [29], we know that the $f - n_i$ last rows of Ψ_f^i can be expressed as a linear combination of its first n_i rows. Then, by multiplying Eq. (40) on the right by $\Pi^{\perp}_{\mathcal{U}_{\nu|\bar{t},f}} \Lambda^{1/2}_{\nu|\bar{t}}$, we get

$$\begin{bmatrix} \mathcal{Y}_{\nu|\bar{t},n_i} \\ \mathcal{Y}_{\nu+n_i|\bar{t}+n_i,f-n_i} \end{bmatrix} \Pi^{\perp}_{\mathcal{U}_{\nu|\bar{t},f}} \Lambda^{1/2}_{\nu|\bar{t}} = \begin{bmatrix} I_{n_i} \\ \mathcal{P}^i \end{bmatrix} \bar{Z}_{\nu|\bar{t}}, \tag{41}$$

where $\mathcal{P}^i$ is defined by $\Psi_f^i(n_i + 1 : f, :) = \mathcal{P}^i \Psi^i_{n_i}$, $\bar{Z}_{\nu|\bar{t}} = \left(\Psi^i_{n_i} \mathcal{X}_{\nu|\bar{t}} \Pi^{\perp}_{\mathcal{U}_{\nu|\bar{t},f}}\right) \Lambda^{1/2}_{\nu|\bar{t}}$, $\Pi^{\perp}_{\mathcal{U}_{\nu|\bar{t},f}}$ is defined as in (21) and $\Lambda_{\nu|\bar{t}} = \operatorname{diag}\left(\lambda^{\bar{t}-\nu}, \ldots, \lambda, 1\right)$, with λ the forgetting factor. Note now that from a theoretical viewpoint, the RQ factorization used in Proposition 2 to implement the orthogonal projection $\Pi^{\perp}_{\mathcal{U}_{\nu|\bar{t},f}}$ is equivalent to multiplying the data equation (40) by $\Pi^{\perp}_{\mathcal{U}_{\nu|\bar{t},f}}$ (see the proof of Proposition 2). Therefore, $\Delta_f(t)$ can simply be obtained as the square of (41), that is,

$$\begin{aligned} \Delta_f(t) = \Sigma(t)(1:f, 1:f) &= \left(\mathcal{Y}_{\nu|\bar{t},f} \Pi^{\perp}_{\mathcal{U}_{\nu|\bar{t},f}} \Lambda^{1/2}_{\nu|\bar{t}}\right) \left(\mathcal{Y}_{\nu|\bar{t},f} \Pi^{\perp}_{\mathcal{U}_{\nu|\bar{t},f}} \Lambda^{1/2}_{\nu|\bar{t}}\right)^\top \\ &= \begin{bmatrix} \bar{Z}_{\nu|\bar{t}} \bar{Z}^\top_{\nu|\bar{t}} & \bar{Z}_{\nu|\bar{t}} \bar{Z}^\top_{\nu|\bar{t}} (\mathcal{P}^i)^\top \\ \mathcal{P}^i \bar{Z}_{\nu|\bar{t}} \bar{Z}^\top_{\nu|\bar{t}} & \mathcal{P}^i \bar{Z}_{\nu|\bar{t}} \bar{Z}^\top_{\nu|\bar{t}} (\mathcal{P}^i)^\top \end{bmatrix}. \end{aligned}$$

By Assumption A5 and Proposition 4 (see Appendix A), it appears clearly that $\operatorname{rank}\left(\bar{Z}_{\nu|\bar{t}}\right) = n_i$. Hence, from the previous equation, we can also see that $\operatorname{rank}(\Delta_f(t)) = \operatorname{rank}(\Delta_{n_i}(t)) = n_i$ before the switch occurs.

After the switch. However, when the system switches at time τ from a submodel i to a submodel j for example, the equation (10) does not hold any longer as there will be in the Hankel matrix formed by the outputs, data generated by two different submodels. Below, we give an illustration of the changes affecting the data equation during the transition from

[4] One can indeed show that when the input is $\mathrm{PE}(f + n_i)$, this relation holds.

i to j. With $\bar{t} = \tau - f$, one can write on a horizon $[\tau - f,\ \tau + f - 1]$

$$\begin{array}{lcccccc}
\tau - 1, & \left[\begin{array}{c} y_f(\bar{t}) \\ \hline -- \end{array}\right] & = & \left[\begin{array}{c} \Gamma^i_f x(\bar{t}) \\ \hline -- \end{array}\right] & + & \left[\begin{array}{c} H^i_f u_f(\bar{t}) \\ \hline -- \end{array}\right] & \\
\tau, & \left[\begin{array}{c} y_{f-1}(\bar{t}+1) \\ \hline y_1(\bar{t}+f) \end{array}\right] & = & \left[\begin{array}{c} \Gamma^i_{f-1} x(\bar{t}+1) \\ \hline \Gamma^j_1 x(\bar{t}+f) \end{array}\right] & + & \left[\begin{array}{c} H^i_{f-1} u_{f-1}(\bar{t}+1) \\ \hline H^j_1 u_1(\bar{t}+f) \end{array}\right] & \\
\vdots & \vdots & = & \vdots & + & \vdots & \\
\tau + f - 2, & \left[\begin{array}{c} y_1(\bar{t}+f-1) \\ \hline y_{f-1}(\bar{t}+f) \end{array}\right] & = & \left[\begin{array}{c} \Gamma^i_1 x(\bar{t}+f-1) \\ \hline \Gamma^j_{f-1} x(\bar{t}+f) \end{array}\right] & + & \left[\begin{array}{c} H^i_1 u_1(\bar{t}+f-1) \\ \hline H^j_{f-1} u_{f-1}(\bar{t}+f) \end{array}\right] & \\
\tau + f - 1, & \left[\begin{array}{c} -- \\ \hline y_f(\bar{t}+f) \end{array}\right] & = & \left[\begin{array}{c} -- \\ \hline \Gamma^j_f x(\bar{t}+f) \end{array}\right] & + & \left[\begin{array}{c} -- \\ \hline H^j_f u_f(\bar{t}+f) \end{array}\right], &
\end{array}$$

where we specify that the time indices $\tau - 1,\ \tau,\ \cdots,\ \tau + f - 1$ (first column of the table) refer respectively to the last times indices of the data vectors $y_f(\bar{t}),\ y_1(\bar{t}+1),\ \cdots,\ y_f(\bar{t}+f)$.

It is an intuitive fact that such a transition shall very likely cause Algorithm 1 to overestimate the order. This is because the mixed data generated by both submodels i and j (assumed to be distinguishable enough) can no longer be fit to one linear model. Therefore, the "nice" structure of the matrix Σ that allowed the extraction of the order is now destroyed. Next, we derive a condition under which the transition (as described above) induced by the switch in the data equation does not, in principle, modify the order estimate provided by Algorithm 1.

Proposition 3. *Let τ be a switching time at which the system switches from a submodel i to a submodel j. Let $\tau_o = \tau - n_i$, $\nu = \tau - \tau_{dwell}$, where τ_{dwell} is the minimum dwell time and let the noise $\{v(t)\}$ be identically null in (32). Assume that*

- *only the submodel i has been active on $[\nu,\ \tau - 1]$,*
- *Assumption A5 holds for a certain $\bar{t} \in [\nu,\ \tau - f + 1]$, where f is strictly greater than all the orders of the system submodels ($\max(n_q) < f \ll \tau_{dwell}$),*
- *Assumption A4 holds for each submodel,*
- *the matrix A_q is nonderogatory for all discrete state value q,*
- *the weight vector γ is such that $\operatorname{rank}\left(\Psi^q_{n_q}\right) = n_q$ for all discrete state value q.*

Then, $\operatorname{rank}\left(\Delta_{n_i+1}(t)\right) = \operatorname{rank}\left(\Delta_{n_i}(t)\right) = n_i$ for all $t \in [\tau,\ \tau + f]$ if and only if

$$\begin{bmatrix} x(\tau_o) \\ u_{2n_i}(\tau_o) \end{bmatrix} \in \operatorname{null}\left(\mathcal{M}^{ij}\right), \tag{42}$$

where $\Delta_r(t)$ is defined as above and $\mathcal{M}^{ij}$ is defined as in (36) and $x(\tau_o) \in \mathbb{R}^{n_i}$. The notation $\operatorname{null}\left(\mathcal{M}^{ij}\right)$ *refers here to the null space of $\mathcal{M}^{ij}$.*

Proof. Let $\tau_o = \tau - n_i$ and $\tau_1 = \tau + n_i$. Since we are interested here only in looking at the rank of $\Delta_{n_i+1}(t)$, where $t \in [\tau,\ \tau + f]$, we can set $f = l = n_i + 1$. Then, consider the system equations written on $[\nu,\ \tau_1]$:

$$\begin{aligned} \mathcal{Y}\Pi_{\mathcal{U}}^{\perp} &= \left[\begin{array}{c|c} \mathcal{Y}_{\nu|\tau_o-1,n_i} & \mathcal{Y}_{\tau_o|\tau-1,n_i} \\ \hline \mathcal{Y}_{\nu+n_i|\tau-1,1} & \mathcal{Y}_{\tau|\tau_1-1,1} \end{array}\right]\Pi_{\mathcal{U}}^{\perp} \\ &= \left[\begin{array}{c|c} \Psi_{n_i}^i \mathcal{X}_{\nu|\tau_o-1} & \Psi_{\text{mix}} \\ \hline \Psi_l^i(l,:)\mathcal{X}_{\nu|\tau_o-1} & \psi_{\text{mix}}^{\top} \end{array}\right]\Pi_{\mathcal{U}}^{\perp} + \left[\begin{array}{c|c} \mathcal{H}_{n_i}^i \mathcal{U}_{\nu|\tau_o-1,n_i} & \mathcal{H}_{\text{mix}} \\ \hline \mathcal{H}_l^i(l,:)\mathcal{U}_{\nu|\tau_o-1,l} & h_{\text{mix}}^{\top} \end{array}\right]\Pi_{\mathcal{U}}^{\perp}, \end{aligned} \tag{43}$$

where

$$\begin{aligned} &\mathcal{U} = \mathcal{U}_{\nu|\tau-1,l} \in \mathbb{R}^{ln_u \times (\tau-\nu)}, \quad \mathcal{Y} = \mathcal{Y}_{\nu|\tau-1,l} \in \mathbb{R}^{l \times (\tau-\nu)}, \\ &\psi_{\text{mix}}^{\top} = \begin{bmatrix} \bar{c}_j^{\top} x(\tau) & \bar{c}_j^{\top} A_j x(\tau) & \cdots & \bar{c}_j^{\top} A_j^{n_i-1} x(\tau) \end{bmatrix}, \\ &h_{\text{mix}}^{\top} = \begin{bmatrix} M_1^j u_1(\tau) & M_2^j u_2(\tau) & \cdots & M_{n_i}^j u_{n_i}(\tau) \end{bmatrix}, \\ &\Psi_{\text{mix}} = \left[\begin{bmatrix} \Psi_{n_i}^i x(\tau_o) \\ -- \end{bmatrix}, \begin{bmatrix} \Psi_{n_i}^i(2:n_i,:)x(\tau_o) \\ \Psi_1^j x(\tau) \end{bmatrix}, \ \ldots, \begin{bmatrix} \Psi_{n_i}^i(n_i,:)x(\tau_o) \\ \Psi_{n_i-1}^j x(\tau) \end{bmatrix}\right], \\ &\mathcal{H}_{\text{mix}} = \left[\begin{bmatrix} \mathcal{H}_{n_i}^i u_{n_i}(\tau_o) \\ -- \end{bmatrix}, \begin{bmatrix} \mathcal{H}_{n_i}^i(2:n_i,:)u_{n_i}(\tau_o) \\ \mathcal{H}_1^j u_1(\tau) \end{bmatrix}, \ \ldots, \begin{bmatrix} \mathcal{H}_{n_i}^i(n_i,:)u_{n_i}(\tau_o) \\ \mathcal{H}_{n_i-1}^j u_{n_i-1}(\tau) \end{bmatrix}\right], \end{aligned}$$

with $M_1^j = \bar{d}_j^{\top}$ and $M_q^j = \begin{bmatrix} \bar{c}_j^{\top} A_j^{q-2} B_j & \cdots & \bar{c}_j^{\top} B_j & \bar{d}_j^{\top} \end{bmatrix}$ for $q \geq 2$. It is important to notice that the last vector of the matrix $\mathcal{Y}$ in Eq. (43) is constructed from time $\tau - 1$ to time $\tau_1 - 1$.

We first show that the first n_i rows of the matrix on the left hand side of (43), namely $\mathcal{Y}\Pi_{\mathcal{U}}^{\perp}$, where $\mathcal{Y} = \mathcal{Y}_{\nu|\tau-1,l}$, are linearly independent. Assumption 5 is assumed to hold for some $\bar{t} < \tau - f + 1$. Then, the data $\mathcal{Y}_{\nu|\bar{t},l}$ and $\mathcal{U}_{\nu|\bar{t},l}$ are generated by a linear model (namely the submodel i), and we can hence use Proposition 4 to conclude that $\begin{bmatrix} \mathcal{Y}_{\nu|\bar{t},l} \\ \mathcal{U}_{\nu|\bar{t},l} \end{bmatrix}$ has $n_i + ln_u$ linearly independent columns. By writing

$$\begin{bmatrix} \mathcal{Y} \\ \mathcal{U} \end{bmatrix} = \left[\begin{array}{c|c} \mathcal{Y}_{\nu|\bar{t},l} & \mathcal{Y}_{\bar{t}+1|\tau-1,l} \\ \mathcal{U}_{\nu|\bar{t},l} & \mathcal{U}_{\bar{t}+1|\tau-1,l} \end{array}\right],$$

it appears clearly that

$$\operatorname{rank}\left(\begin{bmatrix} \mathcal{Y} \\ \mathcal{U} \end{bmatrix}\right) \geq n_i + ln_u.$$

Now by Lemma 2 in [35], we easily get $\operatorname{rank}\left(\mathcal{Y}\Pi_{\mathcal{U}}^{\perp}\right) \geq n_i$. Note that this result holds also for $l = n_i$. Hence the first n_i rows of $\mathcal{Y}\Pi_{\mathcal{U}}^{\perp} \in \mathbb{R}^{l \times (\tau-\nu)}$ are linearly independent. On the other hand, note that[5]

$$\Delta_l(\tau_1 - 1) = \left(\mathcal{Y}\Pi_{\mathcal{U}}^{\perp}\Lambda_{\nu|\tau-1}^{1/2}\right)\left(\mathcal{Y}\Pi_{\mathcal{U}}^{\perp}\Lambda_{\nu|\tau-1}^{1/2}\right)^{\top},$$

where the forgetting weights matrix $\Lambda_{\nu|\tau-1}$ defined as above is nonsingular. Hence, since the square diagonal matrix $\Lambda_{\tau-1}$ is nonsingular, we have $\operatorname{rank}\left(\Delta_l(\tau_1-1)\right) = \operatorname{rank}\left(\mathcal{Y}\Pi_{\mathcal{U}}^{\perp}\right)$. Consequently, we will, in the following, study the rank of $\mathcal{Y}\Pi_{\mathcal{U}}^{\perp}$ instead of that of $\Delta_l(\tau_1-1)$. We are now left with proving that the (n_i+1)-th row of $\mathcal{Y}\Pi_{\mathcal{U}}^{\perp}$ (and therefore that of $\Delta_l(\tau_1-1)$) is a linear combination of its first n_i rows if and if (42) holds. This is equivalent to saying that there exists $\alpha \in \mathbb{R}^{n_i}$ such that

$$\begin{aligned}\Big[\Psi_l^i(l,:)\mathcal{X}_{\nu|\tau_o-1} + \mathcal{H}_l^i(l,:)\mathcal{U}_{\nu|\tau_o-1,l} - \alpha^{\top}\left(\Psi_{n_i}^i\mathcal{X}_{\nu|\tau_o-1} + \mathcal{H}_{n_i}^i\mathcal{U}_{\nu|\tau_o-1,n_i}\right)\Big|& \\ \psi_{\text{mix}}^{\top} + h_{\text{mix}}^{\top} - \alpha^{\top}\left(\Psi_{\text{mix}} + \mathcal{H}_{\text{mix}}\right)\Big]\Pi_{\mathcal{U}}^{\perp} = 0&\end{aligned} \tag{44}$$

Consequently, the matrix that multiplies $\Pi_{\mathcal{U}}^{\perp}$ in (44) can be written as a linear combination of the rows of $\mathcal{U}$. Let $\mathcal{U}$ be partitioned similarly to (43) as $\mathcal{U} = \begin{bmatrix}\mathcal{U}^o & \big| & \mathcal{U}^1\end{bmatrix}$, with $\mathcal{U}^o = \mathcal{U}_{\nu|\tau_o-1,l}$. Then there exists an $r \in \mathbb{R}^{ln_u}$ such that

$$\begin{cases}\Psi_l^i(l,:)\mathcal{X}_{\nu|\tau_o-1} - \alpha^{\top}\Psi_{n_i}^i\mathcal{X}_{\nu|\tau_o-1} + \left(\begin{bmatrix} -\alpha^{\top}\mathcal{H}_{n_i}^i & 0\end{bmatrix} + \mathcal{H}_l^i(l,:)\right)\mathcal{U}^o = r^{\top}\mathcal{U}^o \\ \psi_{\text{mix}}^{\top} + h_{\text{mix}}^{\top} - \alpha^{\top}\left(\Psi_{\text{mix}} + \mathcal{H}_{\text{mix}}\right) = r^{\top}\mathcal{U}^1.\end{cases} \tag{45}$$

By using the Cayley-Hamilton theorem, we have $\Psi_l^i(l,:) = \bar{c}_i^{\top}A_i^{n_i} = -(a^i)^{\top}\Psi_{n_i}^i$, where $a^i = \begin{bmatrix}a_0^i & \cdots & a_{n_i-1}^i\end{bmatrix} \in \mathbb{R}^{n_i}$ is a vector formed with the coefficients of the characteristic polynomial of A_i. Thanks to this property, the first equation of (45) can be rewritten as

$$-\left((a^i)^{\top} + \alpha^{\top}\right)\Psi_{n_i}^i\mathcal{X}_{\nu|\tau_o-1} + \left(\begin{bmatrix} -\alpha^{\top}\mathcal{H}_{n_i}^i & 0\end{bmatrix} + \mathcal{H}_l^i(l,:)\right)\mathcal{U}^o = r^{\top}\mathcal{U}^o. \tag{46}$$

By multiplying this relation by $\Pi_{\mathcal{U}^o}^{\perp}$, we get

$$-\left((a^i)^{\top} + \alpha^{\top}\right)\Psi_{n_i}^i\mathcal{X}_{\nu|\tau_o-1}\Pi_{\mathcal{U}^o}^{\perp} = 0. \tag{47}$$

Since $\Psi_{n_i}^i\mathcal{X}_{\nu|\tau_o-1}\Pi_{\mathcal{U}^o}^{\perp}$ is full row rank, Eq. (47) yields $\alpha = -a^i$. Then, going back to (46), we can derive $r^{\top} = \begin{bmatrix} (a^i)^{\top}\mathcal{H}_{n_i}^i & 0\end{bmatrix} + \mathcal{H}_l^i(l,:)$ because $\mathcal{U}^o$ is full row rank (see Assumption A5). Replacing the expressions of α and r in the second equation of (45), we obtain after some calculations

$$\mathcal{S}^i\Big(\Psi_{n_i}^j x(\tau) + (\mathcal{H}_{n_i}^j - \mathcal{H}_{n_i}^i)u_{n_i}(\tau)\Big) + \mathcal{R}^i\Psi_{n_i}^i x(\tau_o) - \mathcal{S}^i\Psi_{n_i}^i\Omega_{n_i}^i u_{n_i}(\tau_o) = 0,$$

where $\mathcal{R}^i$, $\mathcal{S}^i$, $\Omega_{n_i}^i$ are given by (35). Finally, note that the state $x(\tau)$ can be expressed as

$$x(\tau) = T_{ji}\left(A_i^{n_i}x(\tau_o) + \Omega_{n_i}^i u_{n_i}(\tau_o)\right).$$

[5] We recall that $\tau_1 - 1$ is the last time index of the data involved in (43).

By substituting this expression in the previous equation, the result the condition (42) follows. □

It is easy to verify that when $i = j$, the condition (42) is satisfied for any state $x(\tau_o)$, and any input vector $u_{2n_i}(\tau_o)$. Therefore, for the switches to be detectable on $[\tau, \tau + f]$ by inspecting the rank provided by Algorithm 1, we need to assume that the submodels i and j are different in a certain sense. More precisely,

Assumption A6: The parameters of the constituent submodels of the switched linear system (32) and the input sequence $\{u(t)\}$ are such that for any switch from any submodel i to any other submodel j, the following holds.

$$\begin{bmatrix} x(\tau_o)^\top & u_{2n_i}(\tau_o)^\top \end{bmatrix}^\top \notin \text{null}\left(\mathcal{M}^{ij}\right), \tag{48}$$

where $\mathcal{M}^{ij}$ is defined as in (36) and $x(\tau_o) \in \mathbb{R}^{n_i}$, n_i being the order of the submodel i.

If Assumption A6 holds for all possible switches, then, according to Proposition 3, each switch increases the rank of $\Delta_f(t)$. This means that a change in the dynamics or in the zeros of the system will then be (rigorously without noise, approximately with noise) detectable by the algorithm that estimates the order[6] since such a change induces a rank increase in the matrix $\bar{\Gamma}_f$ (see also [12]). Another way to inspect a change may be for example to keep watch on the variance of the estimates given by the identification process. A switch can then be recognized in that it shall entail a jump in this variance (see [39] for details about abrupt changes detection).

Remark 2. *Note that the identification of both the system matrices and the discrete state (switches detection strategy) are based on the blended output $y_a(t)$ which is the output of the MISO system*

$$\begin{cases} x(t+1) = A_{\mu_{t+1},\mu_t} x(t) + B_{\mu_{t+1},\mu_t} u(t) \\ \quad y_a(t) = \bar{c}_{\mu_t}^\top x(t) + \bar{d}_{\mu_t}^\top u(t), \end{cases} \tag{49}$$

where $\bar{c}_{\mu_t}^\top = \gamma^\top C_{\mu_t}$, $\bar{d}_{\mu_t}^\top = \gamma^\top D_{\mu_t}$. Therefore, there may arise a problem of distinguishability of the submodels. To see this, consider for example two modes i and j with matrices (A_i, B_i, C_i, D_i) and (A_j, B_j, C_j, D_j) in the original system (32) with $A_i = A_j$, $B_i = B_j$ and $C_i \neq C_j$ and $D_i \neq D_j$. Then in (49), these modes are described by $(A_i, B_i, \gamma^\top C_i, \gamma^\top D_i)$ and $(A_j, B_j, \gamma^\top C_j, \gamma^\top D_j)$. However, if $\gamma \in \text{null}(C_i^\top - C_j^\top) \cap \text{null}(D_i^\top - D_j^\top)$, the modes i and j that were different in (32) become indistinguishable in (49). Fortunately, by picking the linear combination weight vector γ at random as suggested by Proposition 1, such degenerate situations can be avoided almost surely. This fact can be proved by following a similar procedure as in the proof of Proposition 1.

Managing the transition period is a rather challenging problem. One issue related to this period is for example the problem of potential state basis change raised in [15]. At each switching time, the state was computed in order to bring the matrices of all the submodels in the same basis. By applying our method for the identification of switched systems, this

[6] even if it is not necessarily followed by an order change

problem is overcome since the matrices of the submodels that have the same order are guaranteed to be in the same basis.

When a switch occurs, pursuing the update (with wrong data) will corrupt the obtained parameters for the submodel i. So, once a switch is detected, the learning of the submodel i needs to be stopped and the final estimates of its parameters need to be recorded. Then, Algorithm 1 can be re-initialized with a new submodel. As there may exist some delay δ between the true switching instant τ and the detected switching time $\hat{\tau}$, the parameters obtained at $\hat{\tau} - \delta$ are recorded instead. In practice, one does not know δ. Fortunately, we have shown in Proposition 3 that this delay is theoretically less or equal to f. Therefore, this presumed delay can be set to verify $\delta \geq f$.

Let $\mathscr{S}$ be the set of submodels that are gradually recorded as they are identified, and let s be a counter of the number of submodels. We formally denote by $M(\mu_t)$ the currently active submodel and we define a vectorized form $\theta(t)$ of the $f+1$ first Markov parameters:

$$\theta(t) = \operatorname{vec}\left(\begin{bmatrix} D(t)^\top & \left(\Gamma_f(t)B(t)\right)^\top \end{bmatrix}^\top\right) \in \mathbb{R}^{(f+1)n_y n_u},$$

where $D(t)$ for example is the matrix D being currently (at time t) estimated and vec is the vectorization operator. By merging two submodels $M(\mu_t)$ and $M(m)$ we will mean, replace the parameters of $M(\mu_t)$ by a normalized weighted sum of the parameters of $M(\mu_t)$ and $M(m)$, i.e. $M(\mu_t) \leftarrow \alpha M(\mu_t) + (1-\alpha)M(m)$ with $0 < \alpha < 1$. Let $d\left(M(\mu_t), M(j)\right)$ be a distance between the submodels $M(\mu_t)$ and $M(j)$. There are many ways to define metrics between dynamical systems (see e.g [40]). For the sake of simplicity, we use here the euclidean distance $d\left(M(\mu_t), M(j)\right) = \sqrt{\left(\theta(t) - \theta(j)\right)^\top \left(\theta(t) - \theta(j)\right)}$ between the vectors of Markov parameters of the submodels $M(\mu_t)$ and $M(j)$. Algorithm 2 estimates the orders and the parameters of each constituent submodel of the switched system (32) while labeling and classifying the different submodels obtained. We recall that this algorithm operates under the conditions of Proposition 3 and additionally under the assumption A6.

5 Numerical examples

5.1 Example 1

In order to illustrate the procedure presented above let us consider a numerical simulation. We consider a switched system resulting from switching among four linear submodels M_1, M_2, M_3, M_4 of order respectively 2, 2, 4, 3.

Algorithm 2 Switches detection and submodels identification and classification

```
1. Initialization: set $\mathscr{S} \leftarrow \emptyset$, $s \leftarrow 1$ and create a submodel $M(\mu_t)$.
2. Update $M(\mu_t)$ using Algorithm 1 until a switch occurs (detected using Proposition 3).
3. Then record $M(\mu_t)$: $\mathscr{S} \leftarrow \mathscr{S} \cup \{M(\mu_t)\}$, and create a new submodel.
4. When the current estimates of parameters have converged, in the sense that $\|\theta(t) - \theta(t-1)\| < \eta_o \theta(t-1)$ for some user-defined threshold $\eta_o > 0$ and for some $t$, then classify the submodel $M(\mu_t)$ as follows. Let $\eta$ be a user-specified threshold, and
       $M(m) = \underset{M(j) \in \mathscr{S}}{\arg\min}\, d\big(M(\mu_t), M(j)\big).$
   If $d\big(M(\mu_t), M(m)\big) < \eta$, then, merge $M(\mu_t)$ and $M(m)$,
   If $d\big(M(\mu_t), M(m)\big) \geq \eta$, then $M(\mu_t)$ is new and so $s \leftarrow s + 1$.
5. Go to step 2 and keep repeating this procedure.
```

$$M_1 : \begin{cases} A_1 = \begin{bmatrix} 0.5386 & -0.5812 \\ -0.5812 & -0.4745 \end{bmatrix}, & B_1 = \begin{bmatrix} 0.7332 & -1.745 \\ 0 & 0.4392 \end{bmatrix} \\ \\ C_1 = \begin{bmatrix} -0 & 0.57762.102 & 0 \end{bmatrix} & D_1 = \begin{bmatrix} 1.432 & 0 \\ 0 & 0 \end{bmatrix}, \end{cases}$$

$$M_2 : \begin{cases} A_2 = \begin{bmatrix} 0.08754 & -0.8488 \\ 0.8488 & 0.08754 \end{bmatrix}, & B_2 = \begin{bmatrix} 0 & -0.9219 \\ 0 & 0 \end{bmatrix} \\ \\ C_2 = \begin{bmatrix} 0.6617 & -0.6294 \\ 0 & 0.01849 \end{bmatrix}, & D_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0.0354 \end{bmatrix}, \end{cases}$$

$$M_3 : \begin{cases} A_3 = \begin{bmatrix} -0.3382 & -0.3114 & 0.598 & -0.1969 \\ -0.465 & -0.5806 & -0.1201 & 0.3968 \\ -0.01017 & -0.3989 & -0.6035 & -0.4613 \\ 0.5263 & -0.3633 & -0.004035 & 0.06349 \end{bmatrix}, & B_3 = \begin{bmatrix} -1.445 & -0.8587 \\ 1.886 & -0.9502 \\ 0.8409 & -0.2533 \\ 0.1275 & 0.9448 \end{bmatrix} \\ \\ C_3 = \begin{bmatrix} -0.5347 & -0.2918 & -0.02547 & 0 \\ 1.583 & -0.8105 & 0.838 & 0.6313 \end{bmatrix}, & D_3 = \begin{bmatrix} 0 & 0.61 \\ 0.6143 & 0 \end{bmatrix}, \end{cases}$$

$$M_4 : \begin{cases} A_4 = \begin{bmatrix} 0.2397 & -0.522 & -0.4535 \\ -0.522 & -0.06531 & -0.4436 \\ -0.4535 & -0.4436 & 0.3772 \end{bmatrix}, & B_4 = \begin{bmatrix} 0 & 0.5096 \\ 2.143 & 0 \\ -0.8066 & -0.6178 \end{bmatrix} \\ \\ C_4 = \begin{bmatrix} 0.3859 & 1.495 & 0 \\ 1.202 & 0.4929 & -0.2108 \end{bmatrix}, & D_4 = \begin{bmatrix} 0 & 0.8659 \\ 0.287 & 0.527 \end{bmatrix}. \end{cases}$$

The input signal is chosen as a white noise of variance unity. The simulation is run with an additive output white noise in the proportion of an SNR = 35 dB. The switching times are 500, 1000, 1500 respectively for the transitions $M_1 \to M_2$, $M_2 \to M_3$, and $M_3 \to M_4$. Prior to the identification process, we show in Table 5.1 that, when driven by the excitation input, the model defined above satisfy the assumption A6. Then, we apply Algorithm 1 with the following set of user-defined parameters : $f = 7 > \max_j(n_j)$, $\lambda = 0.9$ so as to allow a fast convergence while smoothing the parameters estimated; $T_o = 0.1$ is supposed to be slightly greater than the variance σ_v^2 of the noise; the weights vector γ is drawn at random.

In Figure 1 are depicted the order estimate and the magnitude of the estimated model poles versus the time samples. For the readability of this plot, we need to specify that the subsystem M_1 has two real poles; M_2 has a pair of complex poles; M_3 possesses two pairs of complex poles; the poles of M_4 are composed of three real poles. Estimates are in dashed line while true parameters are in solid line.

At each switching time we can notice that the estimate of the order increases suddenly up to f even when there is no change in the order. As a result of a switch, this phenomenon is attributable to the presence of mixed data (generated by two different submodels) in the estimation window. The switching times are detected with a relatively small delay (about f samples at most) in accordance with Proposition 3. The order tracking algorithm detects quickly the switch by jumping to f but its convergence to the real order of the next submodel takes a certain time. This is rather understandable since a convergence is possible only after a certain consistent amount of data has been recorded.

Figure 2 plots the four first Markov Parameters of the model estimated together with that of the actual system. It turns out that the parameters of both the model and the true system are well superposed and that the proposed identification scheme yields good results in the presence of noise. Again, the main interest of this recursive approach is the ability of labeling the different operating modes as they appear, avoiding thus that certain modes go undetected as it may occur in the case of batch identification methods for switched systems.

On the same model given above, and with the same set of tuning parameters, we carried out a Monte-Carlo simulation of size 100 with different realizations of the input and the noise processes. In order to give an illustrative insight of the performances, we plotted in Figure 3 the mean value of the orders and poles estimates obtained then. The results show clearly on our example that the algorithm detects correctly the orders although the detected switching times may vary slightly.

To further test the potentialities of our algorithm, we ran it on a switched system whose

Switch	$M_1 \to M_2$	$M_2 \to M_3$	$M_3 \to M_4$
$\mathcal{M}^{ij} \times \begin{bmatrix} x(\tau_o) \\ u_{2n_i}(\tau_o) \end{bmatrix}$	$\begin{bmatrix} -0.7897 \\ -2.1127 \end{bmatrix}$	$\begin{bmatrix} -1.0523 \\ -0.9373 \end{bmatrix}$	$\begin{bmatrix} 4.4441 \\ 5.6355 \\ 5.5232 \\ 1.2509 \end{bmatrix}$

Table 1: Verification of the assumption A6 for Example 1, driven by a white noise input.

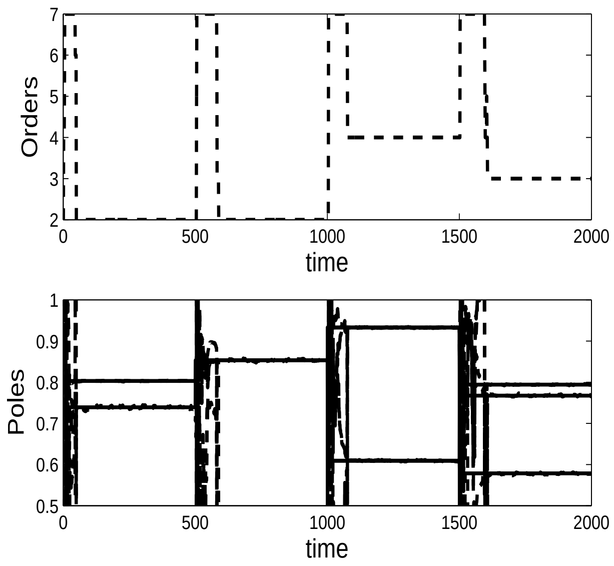

Figure 1: Estimates of the submodels orders and poles (magnitude) obtained on-line. The vertical lines indicate the switching times. The top subfigure represents the estimates obtained for the orders of the submodels. The bottom subfigure represents the magnitude of the submodels poles.

submodels are slowly time-varying. The considered system is composed of the second-order submodels M_1, M_2 whose poles are now varying in magnitude. The variations are created by multiplying the A-matrix of the submodel M_1 by $1 - 5.10^{-2}\sqrt{t - t_1}$ and that of the submodel M_2 by $1 - 10^{-2}\sqrt{t - t_2}$, where $t_1 = 0$ and $t_2 = 1000$. Figure 4 depicts the results obtained. M_1 is active on [0, 1000]; its poles vary slowly in magnitude until they reach zero. In the meantime, the order decreases from 2 to 1 around 400 and increases when a pair of poles becomes again detectable. M_2 is active on [1000, 2000] with much slower variations, the order does not change and the estimates follow almost perfectly the true parameters.

5.2 Example 2

To further test the performance of the proposed method, we consider the simulation example utilized in [15]. In that work, the switching times were assumed to be known and the submodels were assumed to have the same order. Here, neither the switching times, nor the orders (possibly different from a submodel to another) are known. Another difference with the method in [15] is that our algorithm works online. The switched system is composed of four second-order SISO submodels represented by

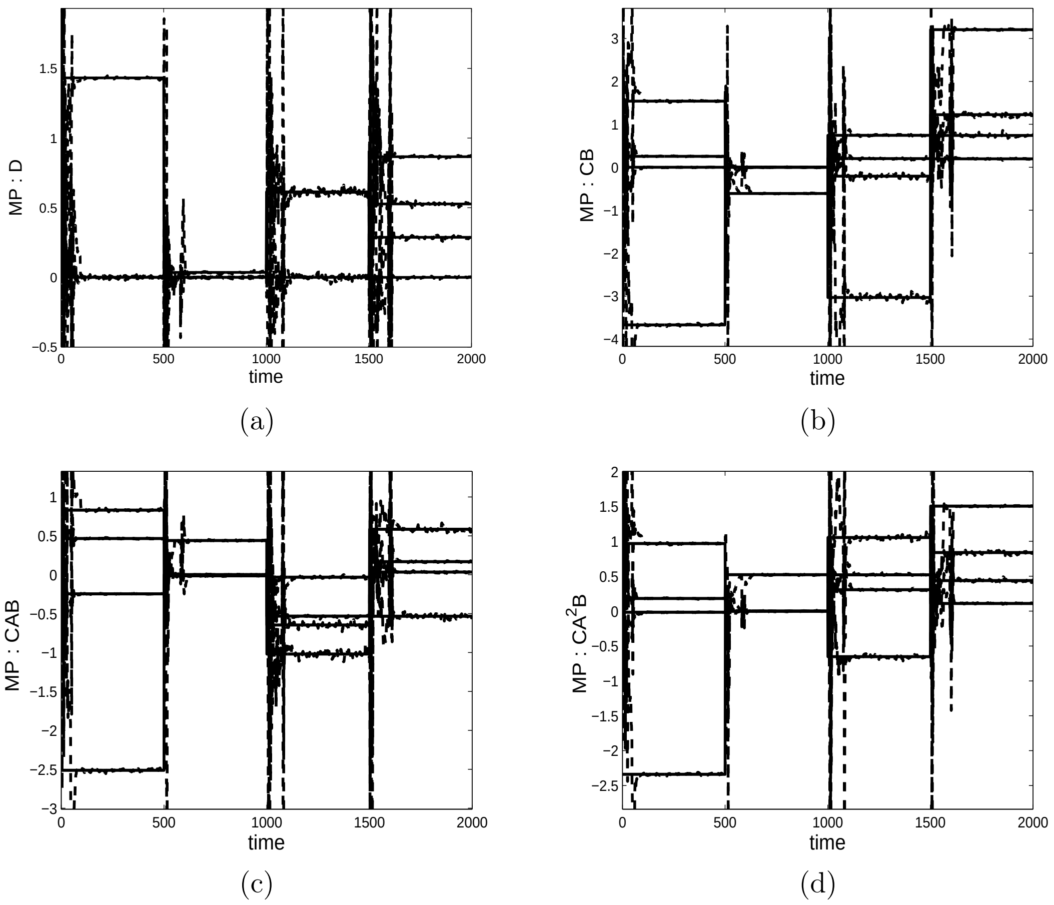

Figure 2: Representation of the four first Markov Parameters (MP) of the real system together with that of the model versus the time samples (a): elements of matrix $D \in \mathbb{R}^{n_y \times n_u}$, (b): $CB \in \mathbb{R}^{n_y \times n_u}$, (c): $CAB \in \mathbb{R}^{n_y \times n_u}$, (d): $CA^2B \in \mathbb{R}^{n_y \times n_u}$.

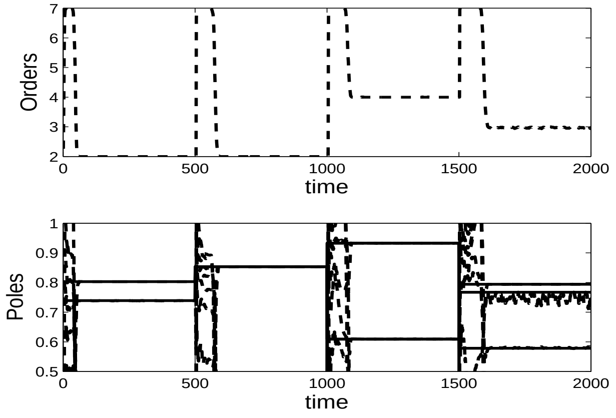

Figure 3: Estimates of the poles and the orders obtained from a Monte-Carlo simulation of size 100.

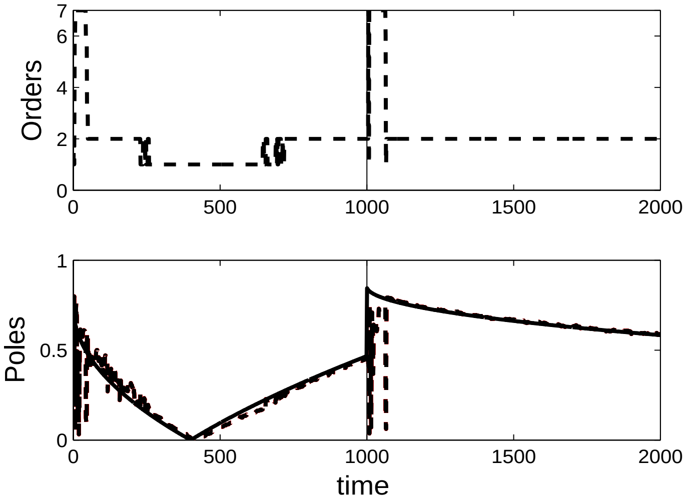

Figure 4: Estimates of the poles and the orders of a switched time-varying system.

$$\left\{\begin{array}{ll} A_1 = \begin{bmatrix} 0 & 0.8 \\ -0.8 & 0.5 \end{bmatrix}, & B_1 = \begin{bmatrix} 0.4 \\ 0 \end{bmatrix}, \\ C_1 = \begin{bmatrix} 1 & 0 \end{bmatrix}, & D_1 = 0, \end{array}\right. \quad \left\{\begin{array}{ll} A_2 = \begin{bmatrix} 0 & 0.5 \\ -0.5 & 0 \end{bmatrix}, & B_2 = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}, \\ C_2 = \begin{bmatrix} 1 & 0 \end{bmatrix}, & D_2 = 0, \end{array}\right.$$

$$\left\{\begin{array}{ll} A_3 = \begin{bmatrix} 0.8 & 0 \\ 0 & -0.3 \end{bmatrix}, & B_3 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \\ C_3 = \begin{bmatrix} 1 & 1 \end{bmatrix}, & D_3 = 0, \end{array}\right. \quad \left\{\begin{array}{ll} A_4 = \begin{bmatrix} 0 & 0.4 \\ -0.4 & 0 \end{bmatrix}, & B_4 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \\ C_4 = \begin{bmatrix} 1 & 0.5 \end{bmatrix}, & D_4 = 0, \end{array}\right.$$

where the B-matrix of the third submodel has been slightly modified to make this latter submodel reachable that is, compatible with our assumptions. We let the exciting input have the same statistical properties as in the previous example and the noise be of the same level. The set of user-defined parameters f, λ, T_o is taken to be roughly the same as in Example 1. Then, a Monte-Carlo simulation of size 100 is carried out with different realizations of the input and the output noise. The results obtained are represented in Figure 5. One can notice that the orders and the poles of the submodels are correctly estimated. The switching times are also well detected based on Proposition 3.

6 Conclusion

This work demonstrates the possibility to identify on-line MIMO linear switched systems in a recursive way using a detection approach. A structured subspace identification scheme has been conveniently prepared to be applied for blindly identifying online the submodels orders and parameters. The switching times are easily recognized since they are followed by an increase in the estimated order. The proposed method may be applied as well to time invariant systems as to slowly time-varying systems. Obviously, some data are inevitably lost in this procedure during the switching transition. In comparison with the existing techniques, the objective here is not so much to fully cluster the data generated by each

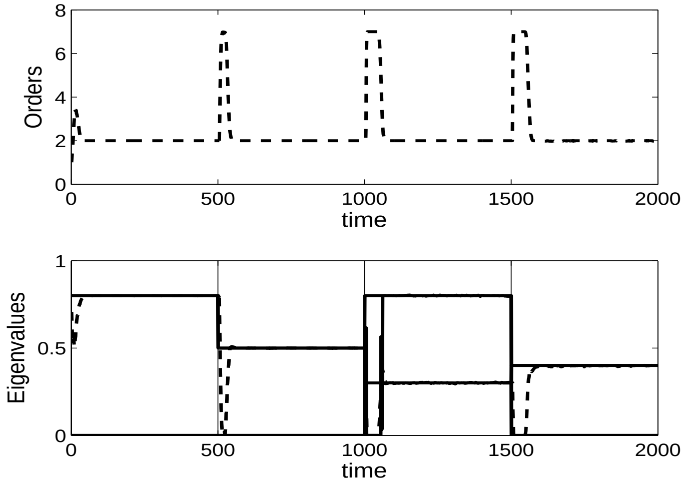

Figure 5: The top subfigure represents the estimates obtained for the orders of the submodels (Example 2). The bottom subfigure represents the magnitude of the submodels poles. The estimates are in dash line while the poles of the true system are in solid. Sometimes only one curve is given per submodel (magnitude of two complex conjugate poles), sometimes two curves are presented for the same submodel (magnitudes of two real poles).

submodel but to obtain instead the submodel of the current operating mode and update its parameters recursively while waiting for any change occurrence. A noteworthy drawback of the off-line methods for switched linear systems identification is that the data basis which is used may often be incomplete so that some functioning modes may be ignored. The scheme proposed here allows to discard this problem but requires unfortunately a certain dwell time.

Appendix A

We state in this Appendix a useful result that is referred to throughout the paper.

Proposition 4. *Assume that the system (1) is reachable and observable and let $v(t)$ be identically null in (1). Consider the input-output data generated by (1) and construct the data matrices $U = U_{1,f,N}$, $Y = Y_{1,f,N}$ and $X = X_{1,N}$ similarly as in (3) with $f \geq n$. Then the following statements are equivalent.*

1. $\operatorname{rank}\left(\begin{bmatrix} X \\ U \end{bmatrix}\right) = n + fn_u$.

2. $\operatorname{rank}(X\Pi_U^{\perp}) = n$, *where*

$$\Pi_U^{\perp} = I_N - U^{\top}(UU^{\top})^{-1}U, \tag{50}$$

I_N *being the identity matrix of order* N.

3. $\operatorname{rank}\left(\begin{bmatrix} Y \\ U \end{bmatrix}\right) = n + fn_u$.

4. rank $\left(Y\Pi_U^\perp\right) = n$.

Since all the signals involved in the data matrices above are assumed to be stochastic, all rank properties should have been stated in the limits. However, these properties should also hold as such for a sufficiently large N.

Appendix B: proof of Proposition 2

For easier manipulations, let us omit the matrices subscripts in the equation (10). Then, we have

$$\bar{Y} = \bar{\Gamma}_f X + \bar{H}_f U + \bar{V}.$$

From the RQ factorization (18), it follows that

$$\bar{Y} = R_{21}Q_1 + R_{22}Q_2 = \bar{\Gamma}_f X + \bar{H}_f R_{11}Q_1 + \bar{V}.$$

Then, multiplying this equation on the left by S and on the right by $Q_2^\top$, one gets

$$SR_{22} = S\bar{Y}Q_2^\top = \begin{bmatrix} I_n \\ \mathscr{P} \end{bmatrix} TXQ_2^\top + S\bar{V}Q_2^\top.$$

since $Q_1Q_2^\top = 0$.
Next, we compute the square of this equation and divide it by N

$$\begin{aligned}\frac{1}{N}SR_{22}R_{22}^\top S^\top = {} & \begin{bmatrix} I_n \\ \mathscr{P} \end{bmatrix} \underbrace{\frac{1}{N}\left(TX\right)\mathcal{Q}\left(TX\right)^\top}_{(I)} \begin{bmatrix} I_n \\ \mathscr{P} \end{bmatrix}^\top \\ & + \underbrace{\frac{1}{N}S\bar{V}\mathcal{Q}\bar{V}^\top S^\top}_{(II)} + S\underbrace{\frac{1}{N}\left(\bar{\Gamma}_f X\mathcal{Q}\bar{V}^\top + \left(\bar{\Gamma}_f X\mathcal{Q}\bar{V}^\top\right)^\top\right)S^\top}_{(III)},\end{aligned} \tag{51}$$

where $\mathcal{Q} = Q_2^\top Q_2 = I - Q_1^\top Q_1 = I - U^\top\left(UU^\top\right)^{-1}U$. Note that the existence of $\mathcal{Q}$ is closely related to the persistency of excitation assumption which states concretely that $UU^\top$ is full rank.
The equation (51) can be simplified as follows:

- Firstly, using the expression of $\mathcal{Q}$, (II) can be written as

$$(II) = S\left(\frac{1}{N}\bar{V}\bar{V}^\top - \frac{1}{N}\bar{V}U^\top\left(\frac{1}{N}UU^\top\right)^{-1}\frac{1}{N}U\bar{V}^\top\right)S^\top.$$

From the independency and ergodicity properties of the process $\{\bar{v}(t)\}$ and the fact that it is statistically uncorrelated with $\{u(t)\}$ (Assumptions A1-A2), the second term in the parentheses appear to be 0 as $N \to \infty$. Then, it remains only that $(II) \to \lim_{N\to\infty}\frac{1}{N}S\left(\bar{V}\bar{V}^\top\right)S^\top = \sigma_v^2\bar{R}_v$ with $\bar{R}_v$ defined as in (22).

- Secondly, (III) will be shown to vanish asymptotically. Notice that

$$\frac{1}{N}\bar{\Gamma}_f X \mathcal{Q} \bar{V}^\top = \frac{1}{N}\bar{\Gamma}_f X \bar{V}^\top - \bar{\Gamma}_f \frac{1}{N} X U^\top \left(\frac{1}{N} U U^\top\right)^{-1} \frac{1}{N} U \bar{V}^\top. \tag{52}$$

Note also that we can write $\bar{V} = D_K V$, where D_K is the block diagonal matrix $D_K = \mathrm{diag}(K(\gamma), \cdots, K(\gamma)) \in \mathbb{R}^{fn_y \times fn_y}$, with $K(\gamma)$ defined as in (9). Then $\lim_{N\to\infty}\left(\frac{1}{N}U\bar{V}^\top\right) = \lim_{N\to\infty}\left(\frac{1}{N}UV^\top D_K^\top\right) = \mathbf{E}\left[u_f(t)v_f(t)^\top\right] D_K^\top = 0$ by Assumption A2. As a consequence, the second term of (52) tends to zero as $N \to \infty$.

Now we need to show that the first term in the right hand side of (52) vanishes asymptotically. To this purpose, we write

$$\frac{1}{N} X \bar{V}^\top = \frac{1}{N}\sum_{t=1}^{f} x(t)\bar{v}_f(t)^\top + \frac{1}{N}\sum_{t=f+1}^{N} x(t)\bar{v}_f(t)^\top. \tag{53}$$

Clearly, $\lim_{N\to\infty}\left(\frac{1}{N}\sum_{t=1}^{f} x(t)\bar{v}_f(t)^\top\right) = 0$ since $\sum_{t=1}^{f} x(t)\bar{v}_f(t)^\top$ is a fixed quantity. To see that $\lim_{N\to\infty}\left(\frac{1}{N}\sum_{t=f+1}^{N} x(t)\bar{v}_f(t)^\top\right) = 0$, let us use Eq. (1) to write

$$\begin{cases} \bar{y}_f(t-f) = \bar{\Gamma}_f x(t-f) + \bar{H}_f u_f(t-f) + D_K v_f(t-f) \\ x(t) = A^f x(t-f) + \Omega_f u_f(t-f), \end{cases} \tag{54}$$

where $\Omega_f = \begin{bmatrix} A^{f-1}B & \cdots & AB & B \end{bmatrix}$. We solve the first equation of (54) for $x(t-f)$ and obtain

$$x(t-f) = \bar{\Gamma}_f^\dagger\big(\bar{y}_f(t-f) - \bar{H}_f u_f(t-f) - D_K v_f(t-f)\big).$$

Then, a combination with the second equation of (54) yields

$$x(t) = A^f \bar{\Gamma}_f^\dagger \bar{y}_f(t-f) + \big(\Omega_f - A^f \bar{\Gamma}_f^\dagger \bar{H}_f\big) u_f(t-f) - A^f \bar{\Gamma}_f^\dagger D_K v_f(t-f).$$

The second term of (53) becomes

$$\begin{aligned} \frac{1}{N}\sum_{t=f+1}^{N} x(t)\bar{v}_f(t)^\top =& A^f \bar{\Gamma}_f^\dagger D_K \Big(\frac{1}{N}\sum_{t=f+1}^{N} y_f(t-f) v_f(t)^\top\Big) D_K^\top \\ &+ \big(\Omega_f - A^f \bar{\Gamma}_f^\dagger \bar{H}_f\big)\Big(\frac{1}{N}\sum_{t=f+1}^{N} u_f(t-f) v_f(t)^\top\Big) D_K^\top \\ &- A^f \bar{\Gamma}_f^\dagger D_K \Big(\frac{1}{N}\sum_{t=f+1}^{N} v_f(t-f) v_f(t)^\top\Big) D_K^\top. \end{aligned} \tag{55}$$

It is a known fact [41] that, under Assumptions A2-A3, the *past* output vector $y_f(t-f)$ is uncorrelated with the *future* noise $v_f(t)$. As a consequence, the first term of Eq. (55) vanishes asymptotically. Moreover, by Assumption A2, the two others terms of Eq. (55) tends to zero as $N \to \infty$. Hence,

$$\lim_{N\to\infty} \frac{1}{N} X\bar{V}^\top = 0 \quad \text{and therefore} \quad \lim_{N\to\infty} (II) = 0.$$

- Finally, by using the expression $\mathcal{Q} = \Pi_U^\perp$, we compute (I) simply as

$$\lim_{N\to\infty} (I) = \lim_{N\to\infty} \frac{1}{N} \mathsf{Z}\Pi_U^\perp \mathsf{Z}^\top, \text{ with } \mathsf{Z} = TX.$$

These three points put together complete the proof.

Acknowledgements. The authors would like to thank the anonymous reviewers whose constructive comments greatly help improve the quality of this manuscrit.

References

[1] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, “A clustering technique for the identification of piecewise affine systems,” *Automatica*, vol. 39, pp. 205–217, 2003.

[2] A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino, “Data classification and parameter estimation for the identification of piecewise affine models,” in *Conference on Decision and Control*, 2004.

[3] R. Vidal, S. Soatto, Y. Ma, and S. Sastry, “An algebraic geometric approach to the identification of a class of linear hybrid systems,” in *Conference on Decision and Control*, 2003.

[4] L. Ljung, *System Identification. Theory for the user.* 2nd ed.. PTR Prentice Hall. Upper Saddle River, USA., 1999.

[5] H. Nakada, K. Takaba, and T. Katayama, “Identification of piecewise affine systems based on statistical clustering technique,” *Automatica*, vol. 41, No 5, pp. 905–913, 2005.

[6] J. Ragot, G. Mourot, and D. Maquin, “Parameter estimation of switching piecewise linear systems,” in *Conference on Decision and Control*, 2003.

[7] A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino, “A bounded-error approach to piecewise affine system identification,” *IEEE Transactions on Automatic Control*, vol. 50, pp. 1567–1580, 2005.

[8] A. L. Juloski, S. Weiland, and W. Heemels, “A bayesian approach to identification of hybrid systems,” *IEEE Transactions on Automatic Control*, vol. 50, No 10, pp. 1520–1533, 2005.

[9] J. Roll, A. Bemporad, and L. Ljung, "Identification of piecewise affine systems via mixed-integer programming," *Automatica*, vol. 40, pp. 37–50, 2004.

[10] Y. Ma and R. Vidal, "Identification of deterministic switched ARX systems via identification of algebraic varieties," in *Hybrid systems computation and control*, 2005.

[11] S. Paoletti, A. Juloski, G. Ferrari-Trecate, and R. Vidal., "Identification of hybrid systems: A tutorial.," *European Journal of Control*, vol. 13,no. 2-3, pp. 242–260, 2007.

[12] J. Borges, V. Verdult, M. Verhaegen, and M. A. Botto, "A switching detection method based on projected subspace classification," in *CDC-ECC*, 2005.

[13] K. Huang, A. Wagner, and Y. Ma, "Identification of hybrid linear time-invariant systems via subspace embedding and segmentation," in *Conference on Decision and Control*, 2004.

[14] K. M. Pekpe, G. Mourot, K. Gasso, and J. Ragot, "Identification of switching systems using change detection technique in the subspace framework," in *Conference on Decision and Control*, 2004.

[15] V. Verdult and M. Verhaegen, "Subspace identification of piecewise linear systems," in *Conference on Decision and Control*, 2004.

[16] Y. Hashambhoy and R. Vidal, "Recursive identification of Switched ARX models with unknown number of models and unknown orders," in *Conference on Decision and Control*, 2005.

[17] R. Vidal, "Recursive identification of switched ARX systems," *To appear in Automatica*, 2008.

[18] L. Bako, G. Mercère, and S. Lecoeuche, "Online subspace identification of switching systems with possibly varying orders," in *European Control Conference*, 2007.

[19] M. Verhaegen and P. Dewilde, "Subspace model identification part 1: The output-error state space model identification class of algorithms," *International Journal of Control*, vol. 56, no. 5, pp. 1187–1210, 1992.

[20] T. Katayama, *Subspace methods for system identification.* Springer-Verlag, 2005.

[21] P. Van Overschee and B. De Moor, *Subspace identification for linear systems. theory, implementation, applications.* Kluwer Academic Publishers, 1996.

[22] M. Lovera, T. Gustafsson, and M. Verhaegen, "Recursive subspace identification of linear and non-linear wiener state-space models," *Automatica*, vol. 36, pp. 1639–1650, 2000.

[23] T. Gustafsson, "Recursive system identification using instrumental variable subspace tracking," in *11th IFAC Symposium on System identification*, 1997.

[24] B. Yang, "An extension of the PASTd algorithm to both rank and subspace tracking," *IEEE Signal Processing Letters*, vol. 2, No. 9, pp. 179–182, 1995.

[25] H. Oku and H. Kimura, "Recursive 4SID algorithms using gradient type subspace tracking," *Automatica*, vol. 38, pp. 1035–1043, 2002.

[26] G. Mercère, S. Lecoeuche, and C. Vasseur, "A new recursive method for subspace identification of noisy systems: EIVPM," in *13th IFAC Symposium on System Identification*, 2003.

[27] G. Mercère, L. Bako, and S. Lecoeuche., "Propagator-based methods for recursive subspace model identification," *Signal Processing*, vol. 88, pp. 468–491, 2008.

[28] J. Munier and G. Y. Delisle, "Spatial analysis using new properties of the cross spectral matrix," *IEEE Transactions on Signal Processing*, vol. 39, pp. 746–749, 1991.

[29] R. A. Horn and C. R. Johnson, *Matrix analysis.* Cambridge University Press, 1985.

[30] J. C. Aguero and G. C. Goodwin, "A recursive algorithm for MIMO stochastic model estimation," in *Asian Control Conference*, 2004.

[31] T. Kailath, *Linear Systems.* Prentice Hall, Inc., Englewood Cliffs, NJ 07632, 1980.

[32] S. Barnett, "Matrices, polynomials, and linear time-invariant systems," *IEEE Transactions on Automatic Control*, vol. Ac-18, no 1, pp. 1–10, 1973.

[33] P. R. Halmos, *Measure Theory.* New York: Springer-Verlag, 1974.

[34] M. Verhaegen, "Subspace model identification part 3: Analysis of the ordinary output-error state space model identification algorithm," *International Journal of Control*, vol. 58, no. 3, pp. 555–586, 1993.

[35] M. Jansson and B. Wahlberg, "On the consistency of subspace methods for system identification," *Automatica*, vol. 34, pp. 1507–1519, 1998.

[36] C. D. Meyer, *Matrix analysis and applied linear algebra.* Cambridge University Press, 2000.

[37] G. Golub and C. V. Loan, *Matrix Computations.* Johns Hopkins, Baltimore, 1996.

[38] M. Petreczky and R. Vidal, "Realization theory of Stochastic Jump-Markov Linear Systems," in *Conference on Decision and Control*, 2007.

[39] M. Basseville and I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application.* Prentice-Hall, Inc., Simon & Schuster Company , Englewood Cliffs, NJ, 1993.

[40] K. De Cock and B. De Moor, "Subspace angles between ARMA models," *Systems and Control Letters*, vol. 46, pp. 265– 270, 2002.

[41] M. Verhaegen, "Identification of the deterministic part of MIMO state space models given in innovations form from input-output data," *Automatica*, vol. 30, pp. 61–74, 1994.