Identification of Switched MIMO ARX models

Laurent Bako[1] and René Vidal[2]

[1] Ecole des Mines de Douai, Département Informatique et Automatique
59508, Douai, France
bako@ensm-douai.fr
[2] Center of Imaging Science, Johns Hopkins University
Baltimore, MD, 21218, USA
rvidal@cis.jhu.edu

Abstract. The paper deals with the identification of a MIMO Switched ARX model from its input-output data. The model is assumed to have an unknown number of submodels of unknown and possibly different orders. This is a challenging problem because of the strong coupling between the unknown discrete state and the unknown model parameters. In our work, we adopt a polynomial decoupling representation for handling switched systems with multiple inputs and multiple outputs. This exact and analytical polynomial representation however comes with an important complexity related to the number of polynomials that need to be estimated. Therefore, an alternative scheme is proposed that operates in two phases. We first classify the data according to the generating submodels and subsequently recover the system parameters.

1 Introduction

A hybrid system can be thought of as a collection of interactive dynamical subsystems with global behavior resulting from switching among all the submodels. The switches can be exogenous deterministic, state-driven, event-driven, time-driven or totally random. Given input-output data generated by such systems, the identification problem aims at determining the parameters that describe the system regardless of the switching mechanism. This problem has received a lot of attention in the modeling-identification community for several reasons. Firstly, the class of piecewise affine models (which is a subclass of hybrid systems) has been shown to possess universal approximation capabilities [?]. Therefore, their relative simplicity and transparency make them very appealing for nonlinear systems modeling and control [?]. In addition, the increasing emergence of discrete event systems in many modern applications has encouraged the modeling, the estimation and the analysis of such systems.

Prior work. Many existing methods for hybrid systems identification apply to piecewise linear systems for which the switching mechanism is defined by regions (polyhedra) of the state-input space. The clustering based procedures [?], [?] first determine these individual regions and then estimate a submodel for each region. The optimization based method [?] solves the identification problem as a linear or quadratic mixed integer programming problem. The algebraic-geometric approach [?] formulates a single decoupling polynomial that vanishes on all the data regardless of their generating submodel. Once this polynomial is computed, the problem boils down to that of recovering the system parameters as the polynomial's derivative evaluated at some specific points [?]. Another category of methods alternates between assigning the data to submodels and estimating simultaneously their parameters by performing a weights learning technique on a fuzzy parameterized model [?], solving a Minimum Partition into Feasible Subsystems (MinPFS) problem [?] or resorting to Bayesian inference [?]. For a comprehensive review of hybrid systems identification techniques, we refer the readers to the survey paper [?]. It is worth noting that most of the aforementioned methods deal with SISO systems with common and available order and sometimes, of known number of submodels. There also exist some works on the identification of linear switched MIMO systems described by state space models [?], [?], [?]. However, the proposed methods generally require the restrictive assumption of a minimum dwell time in each discrete state.

Paper contribution. The contribution of this paper is to consider a general MIMO Switched Auto-Regressive eXogenous (SARX) model where the orders of the submodels are unknown and possibly different, and the number of submodels is not available. The proposed method is based on the concept of Generalized Principal Component Analysis (GPCA) reported in [?]. In contrast to the identification of SISO SARX where only one vanishing homogeneous polynomial is used to embed data lying on a mixture of hyperplanes, the estimation of MIMO SARX involves a potentially unknown number $n_h \geq 1$ of independent homogeneous polynomials that vanish on subspaces that are no longer hyperplanes. In order to conveniently construct the regressors to which the embedding is applied, we first estimate the orders of the submodels and the number of discrete states. Then, given the number of discrete states, we compute the number of vanishing polynomials n_h and subsequently derive the parameters. However, the number of polynomials to be estimated grows exponentially with respect to the number of outputs and the number of submodels, thereby making the computation expensive. Therefore, we propose an alternative method that proceeds by first partitioning the regression data according to each discrete state, based on the estimation of a single polynomial. Given the data related to each mode, the parameters follow from a relevant application of least squares regression techniques.

2 Problem statement

Consider a linear switched system described by the following MIMO ARX model

$$y(t) = \sum_{i=1}^{n_{\lambda_t}} A_{\lambda_t}^i y(t-i) + \sum_{i=0}^{n_{\lambda_t}} B_{\lambda_t}^i u(t-i) + e(t), \tag{1}$$

where $y(t) \in \mathbb{R}^{n_y}$ is the output vector, $\lambda_t \in \{1, \cdots, s\}$ refers to the discrete state, s is the number of submodels of the global switched system and $\left\{A_j^i\right\}_{j=1,\cdots,s}^{i=1,\cdots,n_j} \in$

$\mathbb{R}^{n_y \times n_y}$ and $\left\{B_j^i\right\}_{j=1,\cdots,s}^{i=1,\cdots,n_j} \in \mathbb{R}^{n_y \times n_u}$ are the associated parameter matrices. The modeling error or process noise is represented by $e(t) \in \mathbb{R}^{n_y}$. In this representation, there may exist for certain models j an integer $\delta_j < n_j$ such that $B_j^i = 0$ for $i > \delta_j$ but we require that $A_j^{n_j} \neq 0$ for all j.

Given input-output data $\{u(t), y(t)\}_{t=1}^N$ generated by a switched system of the form (**??**), the identification problem can be formulated as: estimate the number of submodels s, their orders $\{n_j\}_{j=1}^s$ and their parameters $\left\{A_j^i, B_j^i\right\}_{j=1,\cdots,s}^{i=1,\cdots,n_j}$.

3 Identification of switched systems

To begin with the identification procedure, let us define the parameter matrices as

$$\begin{aligned} \mathit{\Gamma}_j &= \left[B_j^{n_j}\ A_j^{n_j}\ \cdots\ B_j^1\ A_j^1\ B_j^0\ A_j^0\right] \in \mathbb{R}^{n_y \times (n_j+1)(n_u+n_y)}, \\ P_j &= \left[0_{n_y \times q_j}\ \mathit{\Gamma}_j\right] \in \mathbb{R}^{n_y \times K},\ j = 1, \cdots, s, \end{aligned} \tag{2}$$

and the regression vector as

$$x_n(t) = \left[u(t-n)^\top\ y(t-n)^\top\ \cdots\ u(t-1)^\top\ y(t-1)^\top\ u(t)^\top\ -y(t)^\top\right]^\top \in \mathbb{R}^K,$$

with $n = \max_j(n_j)$, $A_j^0 = I_{n_y}$, $q_j = (n-n_j)(n_u+n_y)$ and $K = (n+1)\,(n_u + n_y)$. For now, assume that the data is not corrupted by noise i.e., $e(t) = 0$ in (**??**). Then, the equations defining a hybrid system of the form (**??**) may be re-written as

$$(P_1 x_n(t) = 0) \vee \cdots \vee (P_s x_n(t) = 0)\,, \tag{3}$$

where $\vee$ refers to the logic operator *or*. To eliminate the discrete state in this set of $n_y \cdot s$ equations, the underlying idea of the algebraic-geometric method developed in [**?**] is to consider along each row of (**??**), the product of equations related to all the possible submodels. The advantage in doing so is that one can then obtain a set of polynomial constraints $\prod_{j=1}^s \left(\theta_{i_j}^\top x_n(t)\right) = 0$, with $\theta_{i_j}^\top = P_j(i,:)$, satisfied by all the data regardless of their generating submodel. Therefore, the system (**??**) can be decoupled into a set of equations involving n_y^s (not necessarily independent) homogeneous polynomials $p_{i_1,\cdots,i_s}$ defined by

$$\begin{aligned} p_{i_1,\cdots,i_s}(z) = \prod_{j=1}^s \left(\theta_{i_j}^\top z\right) &= \sum h_{i_1,\cdots,i_s}^{n_1,\cdots,n_K} z_1^{n_1} \cdots z_K^{n_K} \\ &= h_{i_1,\cdots,i_s}^\top \nu_s(z), \end{aligned} \tag{4}$$

where $\nu_s(\cdot) : \mathbb{R}^K \to \mathbb{R}^{M_s(K)}$, with $M_s(K) = \binom{K+s-1}{s}$, is the Veronese map which associates to $z \in \mathbb{R}^K$ the vector of all the monomials $z_1^{n_1} \cdots z_K^{n_K}$, $n_1 + \cdots + n_K = s$, organized in a descending lexicographic order. Therefore, each $p_{i_1,\cdots,i_s}$ is a homogeneous polynomial of degree s with coefficients vector $h_{i_1,\cdots,i_s}$ and monomials stacked as a vector in $\nu_s(z) \in \mathbb{R}^{M_s(K)}$.

3.1 Known and equal orders and known number of submodels

In this subsection, we assume that the number of submodels s is known and that the orders of all the submodels are also known and equal to n. Note that the regression vectors $x_n(t)$ generated by the hybrid model (**??**) lie in the union of the subspaces null(P_j) and these subspaces can be estimated using GPCA [**?**]. From the entire set $\{u(t), y(t)\}_{t=1}^{N}$ of input-output data available, if we construct

$$L(n,s) = \big[\nu_s\big(x_n(n+1)\big) \cdots \nu_s\big(x_n(N)\big)\big]^\top \in \mathbb{R}^{(N-n)\times M_s(K)}, \tag{5}$$

then the coefficient vectors $h_{i_1,\cdots,i_s}$ of the vanishing polynomials must satisfy

$$L(n,s)h_{i_1,\cdots,i_s} = 0. \tag{6}$$

Therefore, to solve for the parameters $h_{i_1,\cdots,i_s}$ from (**??**), one needs to compute the null space of the embedded data matrix $L(n,s)$. Note that $h_{i_1,\cdots,i_s}$ is the symmetric tensor product of an indexed set of rows $\left\{\theta_{i_j}\right\}_{j=1}^{s}$ taken from $\{P_j\}_{j=1}^{s}$ respectively: $h_{i_1,\cdots,i_s} = \mathrm{Sym}\left(\theta_{i_1} \otimes \cdots \otimes \theta_{i_s}\right) \in \mathbb{R}^{M_s(K)}$, where $\otimes$ denotes the Kronecker product. It can be shown that these parameters live in a vector subspace of $\mathbb{R}^{M_s(K)}$ that we refer to as the space of homogeneous polynomials of degree s vanishing on the data. In what follows, we define $H = \left[h_1 \cdots h_{n_h}\right]$ to be a basis of this space with dimension n_h.

When the data are perfect and rich enough so that the dimension of the null space of $L(n,s)$ is exactly equal to n_h, the polynomials coefficient matrix H may be computed as a basis of null$(L(n,s))$ by Singular Value Decomposition. The parameters of the system may then be computed by differentiating the polynomials obtained at some particular points. For the sake of completeness, a basic version of the GPCA algorithm [**?**] that computes the system parameters P_j in a deterministic framework, is described in Algorithm **??**. However, in practice the data may be affected by noise. In this case, even with the assumption that the orders and the number of submodels are known, the matrix $L(n,s)$ is likely to be full rank and so, one may not be able to get the right basis H of polynomials. Therefore, it is desirable to know in advance the dimension n_h of this basis. In this way, H could be approximated by the right singular vectors of $L(n,s)$ that correspond to its n_h smallest singular values. But since the P_j are not known, it is not easy to compute n_h in a general framework. However, under some specific assumptions on the intersection between the null spaces of the matrices P_j, we can derive a closed form formula for n_h as outlined in Proposition **??**.

Proposition 1 *Let $B_1, \cdots, B_s$ be some matrices in $\mathbb{R}^{K\times m}$ such that*

(i) *For any $\{i_1, \cdots, i_q\} \subset \{1, \cdots, s\}$, $q \le s$*
$\mathrm{rank}(\left[B_{i_1}, \cdots, B_{i_q}\right]) = \min\Big(K, \sum_{j=1}^{q} \mathrm{rank}(B_{i_j})\Big)$,

(ii) $\sum_{i=1}^{s} \mathrm{rank}(B_i) - s < K$.

Let H be the symmetric tensor product of $B_1, \cdots, B_s$ i.e., H is the matrix whose columns consists of all vectors in $\mathbb{R}^{M_s(K)}$ of the form $\mathrm{Sym}\left(b_{i_1} \otimes \cdots \otimes b_{i_s}\right)$*, where $b_{i_1}, \cdots, b_{i_s}$ are respectively columns of $B_1, \cdots, B_s$.*
Then $\mathrm{rank}(H) = \prod_{j=1}^{s} \mathrm{rank}(B_j)$.

Proof. It can be seen that there exists a matrix $S \in \mathbb{R}^{M_s(K)\times K^s}$ filled with 0 and 1 such that $H = S\,(B_1 \otimes \cdots \otimes B_s)$. A basic result from linear algebra then gives us the result that $\text{rank}(H) \leq \text{rank}(B_1 \otimes \cdots \otimes B_s) = \prod_{j=1}^{s} r_j$, where $r_j = \text{rank}(B_j)$. Let us now exclude the trivial cases and assume none of the B_j is zero. Then, a consequence of the assumption **(i)** is that we can construct some matrices L_j, $j = 1 \cdots , s$ using r_j linearly independent columns of B_j such that $\text{rank}(L) = \min{(r, K)}$, where $L = \begin{bmatrix} L_1, \cdots , L_s \end{bmatrix}$ and $r = \sum_{j=1}^{s} r_j$. Consider now the subset H_L of columns of H defined as $H_L = \text{Sym}(L_1 \otimes \cdots \otimes L_s)$. We will prove by contradiction that $\text{rank}(H_L)$ is in fact $\prod_{j=1}^{s} r_j$. If it was not so, then by denoting columns of H_L by $h_{i_1\cdots i_s}$, there would exist a set of scalars $\lambda_{i_1\cdots i_s}$ not all zero, such that $\sum_{i_1\cdots i_s} \lambda_{i_1\cdots i_s} h_{i_1\cdots i_s} = 0$. Each $h_{i_1\cdots i_s}$ can be regarded as a vector coefficient of the polynomial $p_{i_1\cdots i_s}(x) = \left(b_{i_1}^\top x\right) \cdots \left(b_{i_s}^\top x\right)$, $x \in \mathbb{R}^K$. Therefore we have

$$Q(x) = \sum_{i_1\cdots i_s} \lambda_{i_1\cdots i_s} p_{i_1\cdots i_s}(x) = 0 \quad \forall x. \tag{7}$$

Let $(i_1^o, \cdots , i_s^o)$ be one of the vector of indices such that $\lambda_{i_1^o\cdots i_s^o} \neq 0$. Eq. (**??**) can then be rewritten as

$$Q(x) = p_{i_1^o\cdots i_s^o}(x) + \sum_{i_1\cdots i_s} \lambda^{-1}_{i_1^o\cdots i_s^o} \lambda_{i_1\cdots i_s} p_{i_1\cdots i_s}(x) = 0 \quad \forall\, x. \tag{8}$$

Let $M^o = \begin{bmatrix} L_1^o \cdots L_s^o \end{bmatrix}$ be the matrix L obtained by deleting the columns $b_{i_1^o}, \cdots ,$ $b_{i_s^o}$ from the matrix L. As $\text{rank}(M^o) = r - s < K$, $(M^o)^\perp$, the orthogonal space to the column space of M^o contains at least one nonzero x. For such a x, it is easy to verify that the second term on the right hand side of Eq. (**??**) is zero, and the only term that remains gives us $p_{i_1^o\cdots i_s^o}(x) = \left(b_{i_1^o}^\top x\right) \cdots \left(b_{i_s^o}^\top x\right) = 0$. In

Algorithm 1 GPCA algorithm

```
Step 1: Compute a basis $H$ of the null space of $L(n, s)$ by SVD.
    Let $\mathcal{Q}(z) = \begin{bmatrix} p_1(z) \cdots p_{n_h}(z) \end{bmatrix} = \nu_s(z)^\top H$ be the corresponding basis of homogeneous vanishing polynomials of degree $s$.
Step 2: Differentiate $\mathcal{Q}(z)$ with respect to $z$:
    $\nabla \mathcal{Q}(z) = \begin{bmatrix} \frac{\partial p_1(z)}{\partial z} \cdots \frac{\partial p_{n_h}(z)}{\partial z} \end{bmatrix} = \left( \frac{\partial \nu_s(z)}{\partial z} \right)^\top H$
Step 3: Evaluate $\nabla \mathcal{Q}(z)$ at points $z_j \in \text{null}(P_j)$ but not in $\text{null}(P_i)$, for all $i \neq j$.
Step 4: Then a basis of $\text{null}(P_j)$ can be obtained by SVD as the range space of $\left(\nabla \mathcal{Q}(z_j)\right)^\top$, $j = 1, \cdots , s$. For each $j$, denote the obtained basis by $T_j \in \mathbb{R}^{n_y \times K}$ which must be of dimension $n_y$.
    Let $T_j = \begin{bmatrix} T_j^1 \; T_j^2 \end{bmatrix}$ be a partition of $T_j$ such that $T_j^2 \in \mathbb{R}^{n_y \times n_y}$.
Step 5: $T_j^2$ is necessarily invertible and we can get $P_j = \left(T_j^2\right)^{-1} T_j$, $j = 1, \cdots , s$.
```

other words, $(M^o)^\perp \subset \left(b_{i_1^o}\right)^\perp \cup \cdots \cup \left(b_{i_s^o}\right)^\perp$. In order to show that this cannot hold, we will examine separately the cases 1) $r \leq K$ and 2) $r - s < K < r$.

1) If $r \leq K$, we immediately get a contradiction. To see that, form a full rank square matrix $T = \left[M^o \; B^o \; C\right]$, where B^o is the matrix whose columns are the $b_{i_j^o}$ and C a matrix of appropriate dimensions. T spans $\mathbb{R}^K$ and we can hence find a vector $x \in \mathbb{R}^K$ such that $T^\top x = \left[0 \; 1_s^\top \; c^\top\right]^\top$, where 1_s is a vector of dimension s filled with 1, and c any vector of appropriate dimension. Hence, $x \in (M^o)^\perp$ but $x \notin \left(b_{i_1^o}\right)^\perp \cup \cdots \cup \left(b_{i_s^o}\right)^\perp$.

2) Now examine the case $r - s < K < r$. Let $x \in (M^o)^\perp$, $x \neq 0$. Let k_x be the number of potential vectors $b_{i_j^o}$ obeying $b_{i_j^o}^\top x = 0$ and let $\mathcal{B}_x \in \mathbb{R}^{K \times k_x}$ be a matrix formed by these vectors. Then we have that $0 \leq k_x < K - r + s$ because it would otherwise imply $x = 0$. Therefore, as in the previous case, one can pick a nonzero $y \in (M^o)^\perp$ such that $\mathcal{B}_x^\top y = 1_{k_x}$. It is then easy to see that there exists a point $x_o = y + \alpha x$ for a certain scalar α satisfying $x_o \in (M^o)^\perp$ and $x_o \notin \left(b_{i_1^o}\right)^\perp \cup \cdots \cup \left(b_{i_s^o}\right)^\perp$. We again get a contradiction and the proposition is proved. □

Assumption **(i)** of Proposition **??** corresponds to an important property of the subspace arrangement $\cup_j \operatorname{null}(B_j^\top)$ that is known as transversality. This property states that the dimension of the intersection of any subset of subspaces in the arrangement $\cup_j \operatorname{null}(B_j^\top)$ is as small as possible [**?**]. It is not hard to see that with this assumption, the number of independent homogeneous polynomials that vanish on $\cup_j \operatorname{null}(B_j^\top)$ is equal to $\operatorname{rank}(H)$. Therefore, if the same property holds for $\cup_j \operatorname{null}(P_j)$ and if $(n+1)(n_u + n_y) > (s-1)\, n_y$, then it follows from Proposition **??** that n_h is given by quite a simple formula: $n_h = \prod_{j=1}^{s} \operatorname{rank}(P_j) = n_y^s$ since $\operatorname{rank}(P_j) = n_y$ for all j. Although our formula is less general than the one derived in [**?**], it is much easier to compute. In the rest of the paper we will assume, unless stated otherwise, that the conditions of Proposition **??** hold. In short, given the right n and s, the parameter matrices P_j follow directly from Algorithm **??**. If noise is present in the data, the same algorithm still applies but with the difference that the basis H is approximated by the singular vectors of $L(n, s)$ that are associated with its $n_h = n_y^s$ smallest singular values.

3.2 Unknown number of submodels and unknown and possibly different orders

Now let us consider the challenging case where the orders are possibly different and neither the orders nor the number of submodels are known. Consequently, n_h is also unknown. This means that we need to derive all the characteristics of the switched ARX model (**??**) from the data. In order to properly estimate the parameters of the system, we shall first identify the orders and the number of submodels. Once this task is accomplished, Algorithm **??** can be applied to a certain submatrix of $L(n, s)$ that will be defined later.

Before proceeding further, we need to introduce some notations. For r and l, positive integers, we use the same definitions for $x_r(t)$ and $L(r, l)$ as before. Denote by $n_1 \geq n_2 \geq \cdots \geq n_s$ the orders of the different submodels which constitute the hybrid system and let $\rho = \begin{bmatrix} n_1 & \cdots & n_s \end{bmatrix}$ be a vector consisting of all the orders enumerated in a non-increasing order. Let $\bar{s} \geq s$ and $\bar{n} \geq n_1$ be upper bounds on the number of submodels and their orders respectively. In order to deal with submodels of different orders, we need to introduce the vector of monomials $\eta_\rho\big(x_{n_1}(t)\big)$ that shows up in the product

$$\big(\theta_1^\top x_{n_1}(t)\big) \cdots \big(\theta_s^\top x_{n_s}(t)\big) = h^\top \eta_\rho\big(x_{n_1}(t)\big), \tag{9}$$

of polynomials, where $x_{n_j}(t) \in \mathbb{R}^{K_j}$ and $K_j = (n_j + 1)(n_u + n_y)$. The vector $\eta_\rho\big(x_{n_1}(t)\big)$ is a sub-vector of the vector of monomials $\nu_s\left(x_{n_1}(t)\right)$ that defines the Veronese map and can consequently be obtained by removing some monomials in $\nu_s\left(x_{n_1}(t)\right)$. To this end, we need to define the set of monomials that are to be removed. From the exponents $(\alpha_1, \cdots, \alpha_K)$ of the Veronese map monomials organized in a descending lexicographic order, let us define $\mathscr{I}_\rho$ as the set of indices $(\alpha_1, \cdots, \alpha_K)$ satisfying $\alpha_1 + \cdots + \alpha_j > k_j$ for $j \leq \tau$, where $\tau = K_1 - K_s$ and $k_j = \operatorname{card}\left(\{i : K_i \geq K_1 - j + 1\}\right)$. In the following,

$$V_\rho := \big[\eta_\rho\big(x_{n_1}(\bar{n}+1)\big), \cdots, \eta_\rho\big(x_{n_1}(N)\big)\big]^\top$$

is an embedding data matrix in $\mathbb{R}^{(N-\bar{n}) \times (M_s(K_1) - |\mathscr{I}_\rho|)}$, that is the matrix $L(n_1, s)$ with $|\mathscr{I}_\rho|$ missing columns.

Deriving the system parameters correctly from the data requires that the data satisfy some properties of richness. Therefore, we make the following definition of sufficiency of excitation.

Definition 1. *We say that the data $\{u(t), y(t)\}_{t=1}^N$ are sufficiently exciting for the switched system (**??**) if*

$$\operatorname{rank}(V_\rho) = M_s(K_1) - n_h - |\mathscr{I}_\rho|,$$

where V_ρ is the matrix obtained by removing from $L(n_1, s)$ the columns indexed by $\mathscr{I}_\rho$. That is, the null space of V_ρ is of dimension exactly equal to n_h.

Definition **??** assumes implicitly that all the discrete states have been sufficiently visited. If we denote the matrix of data vectors related to the discrete state j by $\bar{X}_j = \left[x_n(t_1^j) \cdots x_n(t_{N_j}^j)\right]$, where the t_k^j, $k = 1, \cdots, N_j$, are the instants t such that $\lambda_t = j$, then $\bar{X}_j$ must span completely $\operatorname{null}(P_j)$. Otherwise, $\operatorname{null}(P_j)$ may not be identifiable from $\cup_{j=1}^s \operatorname{null}(P_j)$. We have the following result.

Theorem 1 *Assume that the data are sufficiently exciting in the sense of Definition **??**. Assume further that $N_j >> M_{\bar{s}}(K_1)$ for all $j = 1, \cdots, s$. Let $\bar{s} \geq s$ be an upper bound of the number of submodels and r be an integer. Then $\dim\big(\operatorname{null}(L(r, \bar{s}))\big) = 0$ if and only if $r < \max(n_j)$.*

Proof. Assume $r < n_1$. Let q be the number of submodels whose orders are less than or equal to r. Let $\mathcal{X} = \left[x_r(t_1^o), \cdots, x_r(t_{N_o}^o)\right] \in \mathbb{R}^{f \times N_o}$, with $f = (r+1)(n_u + n_y)$, be a matrix of regressors formed by data generated by the $s - q$ submodels of orders $n_j > r$. Then, since the data are diverse enough and are sufficiently exciting, $\mathcal{X}$ has full row rank. It follows from Lemma 5 in [?] that $\text{rank}(\nu_{\bar{s}}(\mathcal{X})) = \min(N_o, M_{\bar{s}}(f)) = M_{\bar{s}}(f)$, where $\nu_{\bar{s}}(\mathcal{X}) = \left[\nu_{\bar{s}}(x_r(t_1^o)), \cdots, \nu_{\bar{s}}\left(x_r(t_{N_o}^o)\right)\right]$. Consequently, $L(r, \bar{s})^\top = \left[\nu_{\bar{s}}(\mathcal{X}), \nu_{\bar{s}}(\mathcal{X}_{s-q+1}), \cdots, \nu_{\bar{s}}(\mathcal{X}_s)\right]$ (up to a column permutation), is necessarily full row rank.

Conversely, if $r \geq \max(n_j)$, the row nullity of each data matrix X_j is at least one. This means that, for all $j = 1, \cdots, s$, there exists a non zero $b_j \in \mathbb{R}^f$ satisfying $b_j^\top \mathcal{X}_j = 0$. One can then verify that $\text{Sym}(b_1 \otimes \cdots \otimes b_s \otimes a_{s+1} \otimes \cdots \otimes a_{\bar{s}}) \in \text{null}(L(r, \bar{s}))$ for some $a_i \in \mathbb{R}^f$. Hence, $\dim(\text{null}(L(r, \bar{s}))) \geq 1$. □

Based on the result stated in Theorem **??**, we shall now be able to estimate the orders along with the number of submodels. The basic fact is that whenever r is less than at least one order n_j, $j = 1, \cdots, s$, there is no polynomial of degree $\bar{s} \geq s$ vanishing on the entire data set embedded in $L(r, \bar{s})$.[3] Therefore, as reported in Algorithm **??**, to obtain for example the first order n_1, set $\rho = \left[r \cdots r\right]$ and hence, $V_\rho = L(r, \bar{s})$. Start decreasing r from $r = \bar{n}$ towards $r = 0$ until $\text{null}(V_\rho) = \{0\}$. Then, fix $n_1 = r + 1$. Once n_1 is available, set $\rho = \left[n_1\, r \cdots r\right]$ and repeat the procedure starting from $r = n_1$ and so on, until all the the orders of all the s submodels are identified. Conversely when all the orders of

[3] Note that we must have $\bar{s} << N$ because there always exists a polynomial of degree equal to N that vanishes on all the data

Algorithm 2 Identification of the orders and the number of submodels

```
Set $j_o \leftarrow 1$, $n_j \leftarrow \bar{n}$ for $j = 1, \cdots, \bar{s}$,
$K \leftarrow (\bar{n} + 1)(n_u + n_y)$, $V \leftarrow L(\bar{n}, \bar{s})$,

1. Determine the maximum order $n_1$ using Theorem ??.
   – While rank($V$) $< M_{\bar{s}}(K)$, do
       • $n_j \leftarrow n_1 - 1$ for $j = 1, \cdots, \bar{s}$
       • $K \leftarrow (n_1 + 1)(n_u + n_y)$
       • $V \leftarrow$ last $M_{\bar{s}}(K)$ columns of $V$
   – EndWhile
   – Obtain the maximum order $n_j \leftarrow n_1 + 1$ for $j = 1, \cdots, \bar{s}$
   – Set $V \leftarrow L(n_1, \bar{s})$
2. Find the remaining orders $n_j$, $j = 2, \ldots, \bar{s}$ using Theorem ??.
   – $j_o \leftarrow j_o + 1$
   – While rank($V$) $< M_{\bar{s}}(K) - |\mathscr{I}_\rho|$
       • $n_j \leftarrow n_{j_o} - 1$ for $j = j_o, \cdots, \bar{s}$
       • Compute $\mathscr{I}_\rho$ and zero out the corresponding columns of $V$
   – EndWhile
   – Obtain the order $n_{j_o}$: $n_j \leftarrow n_{j_o} + 1$ for $j = j_o, \cdots, \bar{s}$
   – Set $V \leftarrow L(n_1, \bar{s})$
3. Go to step 2 until $j_o = \bar{s}$ or until one gets $n_{j_o} = 0$
4. Determine the number of submodels $s = \text{card}(\{j : n_j > 0\})$
```

the s submodels are correctly estimated, r can even go to zero for the $\bar{s}-s$ remaining presumed submodels. If one assumes that $n_j > 0$ for all $j = 1, \cdots, s$, then the number of submodels corresponds to the number of orders n_j strictly greater than zero. One advantage of Algorithm **??** is that it does not require a knowledge of the dimension n_h of the vanishing polynomials space. If all the orders are correctly identified, then Definition **??** of sufficiency of excitation guarantees that the dimension of the null space of V_ρ will then be exactly equal to n_h. Therefore, in Algorithm **??**, we need to compute a basis H_ρ of null(V_ρ) that we should then complete with zeros to form a matrix $H \in \mathbb{R}^{M_s(K_1)\times n_h}$ such that the rows indexed by $\mathscr{I}_\rho$ are null. However the multiple rank tests required may cause Algorithm **??** to fail when dealing with noisy data. Next we shall discuss some possible improvements of the algorithm in order to enhance its ability to deal with noisy data.

3.3 Implementation of Algorithm ??

Recall first that the purpose of the rank tests in Algorithm **??** is to check for a given vector ρ of orders, whether the null space of V_ρ is void or not. Therefore one need not know the rank of V_ρ exactly and just needs a measure of how likely it is that there exists a nonzero vector h_ρ satisfying $V_\rho h_\rho = 0$. Roughly, this can be done by inspecting the smallest singular values of V_ρ for different vectors ρ. For example, to compute n_1, let $\rho^{1,q} = \begin{bmatrix} q \cdots q \end{bmatrix}$, $q = 0, \cdots, \bar{n}$ and define $W_{\rho^{1,q}}$ as the matrix obtained from $\frac{1}{N-\bar{n}} L(\bar{n}, \bar{s})^\top L(\bar{n}, \bar{s})$ by removing its columns and rows indexed by $\mathscr{I}_{\rho^{1,q}}$. Denote by σ_q, $q = 0, \cdots, \bar{n}$ the smallest eigenvalues of the matrix $W_{\rho^{1,q}}$. According to Theorem **??**, $W_{\rho^{1,q}}$ has at least one non zero vector in its null space for all $q \geq n_1$ and hence $\sigma_{n_1} \approx \cdots \approx \sigma_{\bar{n}} \approx \frac{1}{\bar{n}-n_1} \sum_{q=n_1+1}^{\bar{n}} \sigma_q$ and are small compared to $\sigma_0, \cdots, \sigma_{n_1-1}$. Following this procedure, Algorithm **??** can be implemented in a more efficient way for the determination of the orders. With $n_0 = \bar{n}$, $m^1 = \bar{n}$ and given a user-defined decision threshold ε_0, the following algorithm directly computes the orders starting from $j = 1$ through $j = \bar{s}$, by avoiding the rank tests required in Algorithm **??**.

$$
\begin{aligned}
&\rho^{j,q} = \begin{bmatrix} \hat{n}_1 \cdots \hat{n}_{j-1} \, q \cdots q \end{bmatrix}, \; q = 0, \cdots, m^j, \\
&\sigma_q = \min \lambda \left(W_{\rho^{j,q}} \right), \; q = 0, \cdots, m^j, \\
&\varepsilon(r) = \frac{1}{m^j - r} \sum_{q=r+1}^{m^j} \sigma_q, \\
&\mathcal{S}_j = \left\{ r = 0, \cdots, m^j : |\sigma_r - \varepsilon(r)| < \varepsilon_o \right\}, \\
&\hat{n}_j = \begin{cases} \min \left\{ r : r \in \mathcal{S}_j \right\}, & \text{if} \quad \mathcal{S}_j \neq \emptyset \\ m^j & \text{otherwise}, \end{cases} \\
&j \leftarrow j + 1, \\
&m^j \leftarrow \hat{n}_j,
\end{aligned}
$$

where $\lambda(W_{\rho^{j,q}})$ is the set of all eigenvalues of the matrix $W_{\rho^{j,q}}$.

As it turns out, the algorithm derived previously though mathematically accurate, may be computationally prohibitive when the dimensions of the system considered get large. This is worsened by the fact that the regressor $x_{n_1}(t) \in \mathbb{R}^{K_1}$ constructed from all n_y outputs is potentially long and so, it may induce an exponential increase in $M_s(K_1)$, the dimension of homogeneous polynomials space of degree s in K_1 variables. Moreover, as seen above, unless one makes some specific assumptions, the number n_h of polynomials to be estimated is unknown even when the orders and the number of submodels are given. Therefore, we shall discuss in the following an alternative method for dealing with the MIMO SARX system.

4 Complexity reduction using a projection approach

Instead of attempting to compute a potentially large and unknown number of polynomials, we propose in this subsection, a more appealing method to derive the parameters we are looking for. The idea is to use only one decoupling polynomial to partition the data according to the different generating submodels. Once all the data are correctly partitioned, the hybrid system identification problem boils down to a standard regression problem.

Notice that, without loss of generality, system (**??**) can be rewritten, by a straightforward transformation as[4]

$$y(t) = \sum_{i=1}^{n_j} a_j^i y(t-i) + \sum_{i=0}^{n_j} F_j^i u(t-i) + e(t), \tag{10}$$

where the $\left\{a_j^i\right\}_{i=1}^{n_j}$ are the coefficients of the polynomial $z^{n_j} - a_j^1 z^{n_j-1} - \cdots - a_j^{n_j}$ that encodes the poles of the jth submodel as its roots.

Let $\gamma = \begin{bmatrix}\gamma_1 \cdots \gamma_{n_y}\end{bmatrix}^\top$ be a vector of real nonzero numbers and let $y_o(t) = \gamma^\top y(t) \in \mathbb{R}$ be a weighted combination of all the system outputs. Then, (**??**) can be transformed into the following single output system.

$$y_o(t) = \sum_{i=1}^{n_j} a_j^i y_o(t-i) + \sum_{i=0}^{n_j} \gamma^\top F_j^i u(t-i) \tag{11}$$

By introducing the blended output $y_o(t)$ we can obtain only one hybrid decoupling polynomial $P(z)$ that is easier to deal with. But at the same time, the parameters of each mode have been combined and this raises the question if this combination of the outputs preserves the distinguishability of the different submodels that constitute the switched system. In fact, depending on the weights vector γ, two submodels which were initially distinct may reduce to the same submodel in (**??**). To analyse this risk, let

$$F^j = \begin{bmatrix}F_j^{n_j} \cdots F_j^1 \; F_j^0\end{bmatrix}^\top \quad \text{and} \quad a_j = \begin{bmatrix}a_j^{n_j} \cdots a_j^1\end{bmatrix}^\top.$$

[4] Note that the orders n_j in (**??**) may be larger than the ones in (**??**) but we will keep using the same notation

Two modes i and j initially different happen to be indistinguishable consecutively to the previous transformation if they have the same order ($n_i = n_j$), the same dynamics ($a_i = a_j$) and $(F_i - F_j)\gamma = 0$, i.e., when γ lies in $\mathrm{null}(F_i - F_j)$. If the F_j were known one could readily select a γ which does not satisfy this condition. But these matrices are precisely what we are looking for. The question is, without knowing the F_j, how can we choose γ in such a way that for any $i \neq j$, $\gamma \notin \mathrm{null}(F_i - F_j)$.

Lemma 1 *Let $F \neq 0 \in \mathbb{R}^{m \times n_y}$ and $\gamma \in \mathbb{R}^{n_y}$ be a vector of nonzero real numbers generated randomly. Then, we have $F\gamma \neq 0$ with probability one*

Proof. Let $F_k^\top$ be the kth row of F. Let E_k and E be respectively the events $F_k^\top \gamma = 0$ and $F\gamma = 0$. Consider that γ is drawn uniformly from a finite subset of $\mathbb{R} \setminus \{0\}$, of cardinality c. From the Schwartz-Zippel theorem [?], we know that the probability that γ is a root of the one degree polynomial $F_k^\top x$ is bounded as $\Pr(E_k) \leq \dfrac{1}{c}$. Let $r = \mathrm{rank}\,(F)$ and E_{k_i}, $i = 1 \cdots, r$ be the events $F_{k_i}^\top \gamma = 0$ where $\left\{F_{k_i}^\top\right\}_{i=1}^r$ are some r linearly independent rows of F. We have $\Pr(E) = \Pr(E_1 \cap \cdots \cap E_m) = \prod_{i=1}^r \Pr(E_{k_i}) \leq \left(\dfrac{1}{c}\right)^r$. Then, $\Pr\left(\bar{E}\right) \geq 1 - \left(\dfrac{1}{c}\right)^r$ and the proof is completed by letting c tend to infinity. □

From this lemma, it is now clear that if $F_i \neq F_j$, then $(F_i - F_j)\gamma \neq 0$ almost surely for a γ picked randomly. Therefore, two submodels that are distinct in the original system (**??**) remain so after the transformation. However, the separability of the modes, which measures how close the different submodels are, may be affected.

From (**??**), let us redefine the parameter vector $\bar{\theta}_j$ and the regressor $\bar{x}_n(t)$

$$\begin{aligned} \bar{\theta}_j &= \begin{bmatrix} 0_{q_j}^\top & \gamma^\top F_j^{n_j} & a_j^{n_j} & \cdots & \gamma^\top F_j^1 & a_j^1 & \gamma^\top F_j^0 & 1 \end{bmatrix}^\top \in \mathbb{R}^K,\ j = 1, \cdots, s \\ \bar{x}_n(t) &= \begin{bmatrix} u(t-n)^\top & y_o(t-n) & \cdots & u(t)^\top & -y_o(t) \end{bmatrix}^\top \in \mathbb{R}^K, \end{aligned}$$

where $K = (n+1)(n_u+1)$. As previously, we eliminate the dependency of the system equation on the switches by considering the following decoupling polynomial which vanishes on the data independently of their generating submodel:

$$P\big(\bar{x}_n(t)\big) = \prod_{j=1}^{s} \left(\bar{\theta}_j^\top \bar{x}_n(t)\right) = h^\top \nu_s\big(\bar{x}_n(t)\big) = 0. \tag{12}$$

Solving (**??**) is a particular and simpler case ($n_y = 1$) of the case studied in section **??**. The procedure for the determination of $\bar{\theta}_j$ is roughly the same: 1) solve for the orders and number of submodels using Algorithm **??**, 2) obtain h_ρ as any nonzero element in $\mathrm{null}(V_\rho)$ (which is expected to be one dimensional when the data are sufficiently exciting), and 3) complete h_ρ with zeros to form a $h \in \mathbb{R}^{M_s(K)}$ so that the entries of h defined by $\mathscr{I}_\rho$ are zero. Given h, the parameters may be obtained from the derivative of P as shown in [?]:

$$\bar{\theta}_j = \frac{\nabla P(z_j)}{e_K^\top \nabla P(z_j)}, j = 1, \ldots, s, \tag{13}$$

where z_j is a point in $S_j \setminus \cup_{i \neq j}^{s} S_i$, $S_j = \left\{x \in \mathbb{R}^K : \bar{\theta}_j^\top x = 0\right\}$, e_K is a vector of length K with 1 in its last entry and 0 everywhere else.

4.1 Classification of the data

The computation of the $\bar{\theta}_j$ for each submodel, involves the determination of some points lying in S_j but which do not belong to any other S_i, $i \neq j$. The procedure followed is: define $\mathcal{D}_1 = \{t : \nabla P(\bar{x}_n(t)) \neq 0\}$ and for $j > 1$,
$\mathcal{D}_j = \left\{t : \nabla P(\bar{x}_n(t)) \neq 0, \quad \bar{\theta}_i^\top \bar{x}_n(t) \neq 0, i = 1, \cdots, j-1\right\}$. Then, for each submodel, an evaluating point can be determined among the data

$$\tau_j = \arg\min_{t \in \mathcal{D}_j} \left| \frac{\nabla P(\bar{x}_n(t))^\top \bar{x}_n(t)}{e_K^\top \nabla P(\bar{x}_n(t))} \right|$$

from which one can compute the parameters as

$$\bar{\theta}_j = \frac{\nabla P(\bar{x}_n(\tau_j))}{e_K^\top \nabla P(\bar{x}_n(\tau_j))}.$$

We recall that recovering the vectors $\bar{\theta}_j$ which define the blended output $y_o(t)$ is only an intermediate step in achieving the goal of computing the parameters a^j and F^j that define each subsystem of the original system (**??**). Now, from the parameters $\bar{\theta}_j$ obtained, we can determine the discrete state of (**??**) which is the same as that of (**??**) and then, compute finally the system sought. In order to discard possible outliers in the data we set up a performance bound $\varepsilon < 1$ to define the following decision rules:

$$\text{If} \quad \min_j \Delta(\bar{\theta}_j, \bar{x}_n(t)) > \varepsilon \|\bar{x}_n(t)\|, \quad \text{then} \quad \lambda_t \quad \text{is undecidable} \tag{14}$$

$$\text{If} \quad \min_j \Delta(\bar{\theta}_j, \bar{x}_n(t)) \leq \varepsilon \|\bar{x}_n(t)\|, \quad \text{then} \quad \lambda_t = \arg\min_j \Delta\left(\bar{\theta}_j, \bar{x}_n(t)\right) \tag{15}$$

where $\Delta(\bar{\theta}_j, \bar{x}_n(t)) = \dfrac{\left|\bar{\theta}_j^\top \bar{x}_n(t)\right|}{\|\bar{\theta}_j\|}$ is the distance from the point $\bar{x}_n(t)$ to the linear hyperplane S_j defined by its normal vector $\bar{\theta}_j$. We define $\mathscr{X}_j = \{t > \bar{n} : \lambda_t = j\}$ $= \left\{t_1^j, \cdots, t_{N_j}^j\right\}$, $j = 1, \cdots, s$ as the set of time instances corresponding to the data generated by the mode j.

4.2 Estimation of the submodels

Based on the results of the previous classification, we know the data corresponding to each generating mode. Therefore, it remains to determine the parameters of each mode j from the data indexed by $\mathscr{X}_j$. To begin with, consider a single linear submodel j of order n_j from (**??**). For any $t \in \mathscr{X}_j$, let us define

$$\begin{aligned} \Phi_j^y(t) &:= \left[y(t-1) \cdots y(t-n_j)\right] \in \mathbb{R}^{n_y \times n_j}, \\ \phi_j^u(t) &:= \left[u(t)^\top \cdots u(t-n_j)^\top\right]^\top \in \mathbb{R}^{(n_j+1)n_u}, \end{aligned}$$

and $z_j(t) := y(t) - \Phi_j^y(t)a_j$ and $\phi_j^o(t) := \Phi_j^y(t)^\top \gamma$. Then, Eqs. (**??**) and (**??**) can be written as

$$\begin{aligned} z_j(t) &= F_j \phi_j^u(t) + e(t), \\ y_j^o(t) &= a_j^\top \phi_j^o(t) + \gamma^\top F_j \phi_j^u(t) + \gamma^\top e(t). \end{aligned}$$

Repeating these equations for all the available data leads to the following block equations

$$Z_j = F_j U_j + E_j, \tag{16}$$

$$Y_j^o = a_j^\top \Psi_j^o + \gamma^\top F_j U_j + \gamma^\top E_j, \tag{17}$$

where

$$U_j = \left[\phi_j^u(t_1^j) \cdots \ \phi_j^u(t_{N_j}^j)\right], \ Z_j = \left[z_j(t_1^j) \cdots \ z_j(t_{N_j}^j)\right], \ E_j = \left[e(t_1^j) \cdots \ e(t_{N_j}^j)\right],$$

and

$$Y_j^o = \left[y_j^o(t_1^j) \cdots \ y_j^o(t_{N_j}^j)\right], \ \Psi_j^o = \left[\phi_j^o(t_1^j) \cdots \ \phi_j^o(t_{N_j}^j)\right].$$

To solve for a from (**??**), we multiply on the right side by

$$\Pi_{U_j}^\perp = I - U_j^\top \left(U_j U_j^\top\right)^{-1} U_j$$

to eliminate the term $\gamma^\top F_j$ that is not required. This results in

$$Y_j^o \Pi_{U_j}^\perp = a_j^\top \Psi_j^o \Pi_{U_j}^\perp + \gamma^\top E_j \Pi_{U_j}^\perp. \tag{18}$$

and therefore, we get

$$\hat{a}_j^\top = \arg\min_a \left\| Y_j^o \Pi_{U_j}^\perp - a^\top \Psi_j^o \Pi_U^\perp \right\|_F^2 = Y_j^o \Pi_U^\perp \Psi_j^{o\top} \left(\Psi_j^o \Pi_{U_j}^\perp \Psi_j^{o\top}\right)^{-1}.$$

Once $\hat{a}_j$ is available, an estimate of $z_j(t)$ can be obtained according to $\hat{z}_j(t) = y(t) - \Phi_j^y(t)\hat{a}_j$ from which the matrix $\hat{Z}_j$ defined above is constructed. We get finally the matrix F_j as,

$$\hat{F}_j = \arg\min_{F_j} \left\| \hat{Z}_j - F_j U_j \right\|_F^2 = \hat{Z}_j U_j{}^\top \left(U_j U_j{}^\top\right)^{-1}.$$

The estimates obtained here result from a classification task where only a finite amount of data has been processed. This raises the question whether the $\hat{a}_j^\top$ and $\hat{F}_j$, $j = 1 \cdots, s$, would converge to the true parameters if the dataset used for the identification was infinite (each mode is visited an infinite number of times). This can be easily proved by assuming that all the data are correctly partitioned and that the noise $e(t)$ is white. However, to rightly partition an infinite amount of data, one may need to resort to recursive identification.

5 Numerical results

In order to test the performances of the method presented in the paper, consider the following linear switched system with 1 input and 2 outputs and composed of 2 submodels of orders 2 and 1. The system is driven by a zero-mean white Gaussian noise input with unit standard deviation and switches periodically from one discrete state to another every 10 samples. We add to the data an additive output noise in the proportion of a signal to noise ratio SNR = 30 dB.

$$y(t) = a_j^1 I_{n_y} y(t-1) + a_j^2 I_{n_y} y(t-2) + b_j^0 u(t) + b_j^1 u(t-1) + b_j^2 u(t-2),$$

where a_j^1 and a_j^2, $j = 1, 2$ are scalar coefficients and b_j^0, b_j^1, b_j^2 are vectors of dimension $n_y = 2$. Of course, in this representation, a_j^2 and b_j^2 are zero for the second submodel. For convenience of comparison with the estimates, we present the system parameters with the following matrices:

$$\mathit{\Gamma}_1 = \left[\begin{array}{cc|cc|c|c|c} 1.3561, & 0 & 0.6913, & 0, & 0 & 0.3793 & 0.2639 \\ 0, & 1.3561 & 0, & 0.6913, & 1.3001 & 1.8145 & 0.7768 \end{array}\right],$$

$$\mathit{\Gamma}_2 = \left[\begin{array}{cc|cc|c|c|c} 0.9485, & 0 & 0, & 0, & 1.7661 & 2.9830 & 0 \\ 0, & 0.9485 & 0, & 0, & 0 & 0.9106 & 0 \end{array}\right],$$

defined with respect to the regression vector

$$\left[y(t-1) \,|\, y(t-2) \,|\, u(t) \,|\, u(t-1) \,|\, u(t-2) \right].$$

As described above, given input-output data generated by this system on a time window of size 1500, we are interested in extracting the number of the constituent submodels, the orders of these submodels and the parameters that describe them. To demonstrate the performances of our algorithm we carried out a Monte-Carlo simulation of size 1000 with the following user-defined set of parameters : $\bar{n} = 3$, $\bar{s} = 3$. For a threshold of $\varepsilon_0 = 10^{-3}$ in the algorithm of subsection **??**, the estimation of the orders of both submodels is realized with 100% of successes. Since we provided $\bar{s} = 3$, the vector of orders is obtained as $\hat{\rho} = \left[2\ 1\ 0\right]$. The means of the estimates $\hat{\mathit{\Gamma}}_1$ and $\hat{\mathit{\Gamma}}_2$ obtained across all the simulations are given below:

$$\hat{\mathit{\Gamma}}_1 = \left[\begin{array}{cc|cc|c|c|c} 1.3558, & 0.0043 & 0.6897, & 0.0036 & 0.0056 & 0.3937, & 0.2639 \\ -0.0012,, & 1.3558 & -0.0021, & 0.6907 & 1.3031 & 1.8208 & 0.7753 \end{array}\right]$$

$$\hat{\mathit{\Gamma}}_2 = \left[\begin{array}{cc|cc|c|c|c} 0.9480, & 0.0045 & -0.0005, & 0.0050 & 1.7710, & 2.9869, & 0.2695 \\ -0.0003, & 0.9479 & -0.0001, & -0.0006 & -0.0012 & 0.9081 & -0.0018 \end{array}\right]$$

In Figure **??**, we present the histogram of the maximum angle between the column space of the hybrid parameter matrix H and that of its estimate $\hat{H}$. It can be noticed that for all the simulations the cosine of this angle is larger than 0.99 which implies a strong correlation between H and its estimate. In the case with projection, this result is much better since H consists of only one vector. In Figure **??**, we present the absolute errors between the true parameter matrices P_j and their respective estimates $\hat{P}_j$ obtained by our algorithm. It turns out that the proportion of simulations that give errors less than 0.1 is about 50% for

(a) (b)
GPCA GPCA+classification

Fig. 1: Histograms of the maximum subspace angle between span(H) and span($\hat{H}$).

(a) (b)
GPCA GPCA+classification

Fig. 2: Histograms of the errors $\left\|P_1 - \hat{P}_1\right\|_2$ and $\left\|P_2 - \hat{P}_2\right\|_2$
.

the first model and about 70% for the second submodel. These proportions are significantly improved (92% and 94%) when we use the classification approach described in section **??**.

6 Conclusion

We have presented an algebraic-geometric method for the identification of MIMO SARX models with unknown number of submodels of unknown and possibly different orders. Given upper bounds on the number of submodels and on the orders, the strategy followed consists of computing first the right number of submodels and the orders. We then showed that, with sufficiently exciting data the parameters can be extracted by applying the concept of GPCA. Finally, we provided some simulations results that show the potential of this method. Future work includes extending the work presented in [**?**] on recursive identification of SISO switched systems to MIMO SARX.