Basis transform in switched linear system state-space models from input-output data

Fethi Bencherki *  
Semiha Türkay †  
Hüseyin Akçay ‡

August 13, 2021

Abstract

This paper tackles the basis selection issue in the context of state-space hybrid system identification from input-output data. It is often the case that an identification scheme responsible for state-space switched linear system (SLS) estimation from input-output data operates on local levels. Such individually identified local estimates reside in distinct state bases, which call for the need to perform some basis correction mechanism that facilitates their coherent patching for the ultimate goal of performing output predictions for predefined input test signals. We derive necessary and sufficient conditions on the submodel set, the switching sequence, and the dwell times that guarantee the presented approach’s success. Such conditions turn out to be relatively mild, which contributes to the application potential of the devised algorithm. We also provide a linkage between this work and the existing literature by providing several insightful remarks that highlight the discussed method’s favorability. We supplement the theoretical findings by an elaborate numerical simulation that puts our methodology into action.

AMS: 93B30, 93B15, 93C30, 93C05 93C95.

Keywords: Hybrid system, switched linear system, state-space model, linear time-varying, basis transform, graph theory.

1 Introduction

Hybrid systems characterize interactions between the discrete and the continuous phenomenons. Such systems have seen massive application scope recently due to their potential for modelling nonlinear dynamics. Hybrid systems have applications in numerous areas of great interest to both scientific and industrial communities, for instance, in chemical processes [1], robotics manufacturing processes [2, 3], air traffic systems [4, 5, 6], computer and communication networks [7, 8, 9, 10], power systems [11], medical applications [12], human control behavior [13], computer vision [14], and biology and environmental systems [15].

The state-space SLSs are one important class of hybrid systems governed by an external arbitrary switching sequence. Operating in the state-space framework has its apparent privileges as important structural properties like observability and controllability that could be checked straightforwardly. Other important tasks like fault detection and observer design could be carried out relatively easily compared to the input-output models. One extra challenge emerging from the identification of such systems is that the

*Department of Electrical and Electronics Engineering, Eskişehir Technical University, 26555 Eskişehir, Turkey. E-mail: fethi.bencherki@eskisehir.edu.tr
†Department of Electrical and Electronics Engineering, Eskişehir Technical University, 26555 Eskişehir, Turkey. E-mail: semihaturkay@eskisehir.edu.tr
‡Corresponding author. Department of Electrical and Electronics Engineering, Eskişehir Technical University, 26555 Eskişehir, Turkey. E-mail: huakcay@eskisehir.edu.tr
identified discrete states, when identified individually, are present in a different state basis \[16\]. Then, the identified SLS can not be used for conducting output predictions.

Often, models obtained from the practice of system identification are deployed in some end applications ranging from analysis to obtain some information about the system to control purposes. However, such objectives, more or less, require performing simulations on the obtained model. Being a rich model set, the SLSs pertain as well to such scenarios and end-uses. When arisen in the SLS modelling context, such facts and necessities invoke the basis issue.

The basis issue in the SLSs could be circumvented by considering canonical representations. Transformation of arbitrary state-space models to the observable canonical form was studied in the literature for single-input/single-output (SISO) or multi-input/single-output (MISO) systems \[17\]. For this approach to be applicable, the original system must be in a canonical form. Therefore, it suffices to transform the identified discrete states to a canonical form to lift the basis issue. For an observable multi-input/multi-output (MIMO) system with characteristic polynomial equal to minimal polynomial, an observability canonical form can be retrieved following the procedure in \[18\]. The idea is to replace one output of the system with a linear combination of other outputs so that a linear transformation to the observability canonical form can be found by randomization. In \[19\], discrete states were estimated on non-overlapping segments from input-output data by a subspace algorithm and brought to a common basis by similarity transformations. A key role is played by the observability matrices computed from input-output data. This paper revisits the basis transform problem in \[19\] and derives further and detailed results.

The paper is organized in the following fashion. Section 2 formulates the basis transformation problem. In Section 3, a solution to this problem is provided and an algorithm to compute the transformation from input-output data is outlined. Section 4 is devoted to a numerical study of the proposed algorithm. A central theme in this paper is the efficient use of input-output data and switches for basis construction. Necessary and sufficient conditions are put forward and the derived results are interpreted from graph theory perspective.

2 Problem formulation

Let a discrete-time, linear time-varying, MIMO system be given by the state-space equations

\[
\begin{align*}
x(k+1) &= A(k)x(k) + B(k)u(k), \\
y(k) &= C(k)x(k) + D(k)u(k)
\end{align*}
\]

where \(u(k) \in \mathbb{R}^m, y(k) \in \mathbb{R}^p\), and \(x(k) \in \mathbb{R}^n\) are respectively the input, the output, and the state sequences. We assume \(n\) is known and does not change with time. Let \(\phi\) be a switching sequence from the set of positive integers denoted by \(\mathbb{N}\) onto a finite set \(\mathbb{S} = \{1, \cdots, \sigma\}\) for some fixed \(\sigma\). Set \(j = \phi(k)\) and suppose that \(A(k) = A_j, B(k) = B_j, C(k) = C_j, D(k) = D_j\) where \(\mathcal{P}_j = (A_j, B_j, C_j, D_j) \in \mathcal{P}, j = 1, \cdots, \sigma\). Together with \(\phi\), \((1) - (2)\) form a switched linear system.

A switching sequence \(\phi(k)\) segments \([1,N]\) into disjoint intervals \([k_i, k_{i+1}]\) such that \(\phi(k) = \phi(k_i)\) for all \(k_i \leq k < k_{i+1}\) where \(k_0 = 1\) and \(k_i < k_p \leq N\). Given a segmentation of \([1,N]\) denoted by \(\chi\), we let \(\delta_i(\chi) = k_{i+1} - k_i, 0 \leq i < i^*\) for the dwell time of the discrete state active in \([k_i, k_{i+1})\). We assume that \(\mathcal{P}_j, 1 \leq j \leq \sigma\) are stable and minimal. The problem of basis transform from input-output data is as follows.

**Problem 2.1** Given \(\mathcal{P}_i \sim \mathcal{P}_{\phi(k_i)}, 0 \leq i \leq i^*\) and \(u(k), y(k), 1 \leq k \leq N, \) find \(\mathcal{P}_j = (A_j, B_j, C_j, D_j), 1 \leq j \leq \sigma\) preserving the input-output map and \(\mathcal{P}_{\phi(k_i)} \sim \mathcal{P}_{\phi(k_i)}, 0 \leq i \leq i^*\).
identified submodels are only similar to the true ones most often, this approach does not admit correct input-out maps unless a basis transform is performed on the identified submodels. In Section 3, we provide a solution to Problem 2.1 and illustrate the basis transform procedure proposed in this paper by a numerical example in Section 4.

3 Basis transformation in switched linear systems

Fix $i$ and let $j_1 = \phi(k_{i-1})$ and $j_2 = \phi(k_i)$. If $\mathcal{P}_j$ is similar to $\hat{\mathcal{P}}_j$, this relationship is shown by $\mathcal{P}_j \sim \hat{\mathcal{P}}_j$. For some invertible matrices $T_{j_1}$ and $T_{j_2}$, the following relationships

\[ A_{j_1} = T_{j_1}^{-1} \hat{A}_{j_1} T_{j_1}, \quad \hat{C}_{j_1} = \hat{C}_{j_1} T_{j_1}, \quad B_{j_1} = T_{j_1}^{-1} \hat{B}_{j_1}, \quad D_{j_1} = \hat{D}_{j_1} \]

\[ A_{j_2} = T_{j_2}^{-1} \hat{A}_{j_2} T_{j_2}, \quad \hat{C}_{j_2} = \hat{C}_{j_2} T_{j_2}, \quad B_{j_2} = T_{j_2}^{-1} \hat{B}_{j_2}, \quad D_{j_2} = \hat{D}_{j_2} \]

hold. The Markov parameters are matched by equating $D(k)$ to $\hat{D}(k)$ at $k_{i-1}$ and $k_i$ and for lags $l \geq 1$ by

\[ C_{j_1} A_{j_1}^{l-1} B_{j_1} = \hat{C}_{j_1} \hat{A}_{j_1}^{l-1} \hat{B}_{j_1}, \quad C_{j_2} A_{j_2}^{l-1} B_{j_2} = \hat{C}_{j_2} \hat{A}_{j_2}^{l-1} \hat{B}_{j_2}. \]

(4) Since $j_1, j_2 \in \mathbb{S}$, there are $\sigma$ distinct equations (3–4). The number of the distinct pairs $(j_1, j_2) \in \mathbb{S}^2$ is at most $\sigma(\sigma - 1)$.

3.1 Estimation of the initial states

Let $q_{i-1} = \eta_{i-1} - \xi_{i-1} > 0$ and $\xi_{i-1}, \eta_{i-1} \in [k_{i-1} k_i)$. Recursively using (1–2), we derive

\[ Y_{i-1} = \mathcal{O}(j_1) x(\xi_{i-1}) + \Gamma(j_1) U_{i-1} \]

(5) where

\[ Y_{i-1} = [y^T(\xi_{i-1}) \cdots y^T(\eta_{i-1})]^T, \]

\[ U_{i-1} = [u^T(\xi_{i-1}) \cdots u^T(\eta_{i-1})]^T, \]

\[ \mathcal{O}(j_1) = \begin{bmatrix} C_{j_1} \\ \vdots \\ C_{j_1} A_{j_1}^{q_{i-1}-1} \end{bmatrix}, \]

(6)

\[ \Gamma(j_1) = \begin{bmatrix} D_{j_1} & 0 \\ \vdots & \ddots & \vdots \\ C_{j_1} A_{j_1}^{q_{i-1}-1} B_{j_1} & \cdots & D_{j_1} \end{bmatrix}. \]

(7)

Let $\hat{O}(j_1)$ and $\hat{\Gamma}(j_1)$ denote the extended observability and the lower triangular block Toeplitz matrices generated by $\hat{\mathcal{P}}_{j_1}$ similarly to (6–7). Note that $\mathcal{O}(j_1) = \hat{O}(j_1) T_{j_1}$ and $\Gamma(j_1) = \hat{\Gamma}(j_1)$. Substitute $\hat{x}(\xi_{i-1}) = T_{j_1} x(\xi_{i-1})$ and denote the Moore-Penrose pseudo inverse of a given full column-rank matrix $X$ by $X^\dagger = (X^T X)^{-1} X^T$. If $q_{i-1} \geq n - 1$, from minimality of $\hat{\mathcal{P}}_{j_1}$ and (5), we determine $\hat{x}(\xi_{i-1})$ uniquely

\[ \hat{x}(\xi_{i-1}) = \hat{O}(j_1)^\dagger (Y_{i-1} - \Gamma(j_1) U_{i-1}). \]

(8) Since $q_{i-1} > 0$, (8) is valid if $n \geq 2$ which we assume in the sequel. The case $n = 1$ is uninteresting.

Since $\xi_{i-1} = k_{i-1}$ and $\eta_{i-1} = k_i - 1$ are feasible and $q_{i-1} \geq n - 1$, we must require $\delta_{i-1}(X) \geq n$. Note that $k_{i-1} \leq \xi_{i-1} \leq k_i - n$. 

3
3.2 Input-output preserving transformation

Recursively using (1)–(2), we derive for all $k \in [k_l, k_{i+1})$
\[
y(k) = C(k)\Phi(k, \xi_{i-1})x(\xi_{i-1}) + \sum_{l=\xi_{i-1}}^{k} h(k, l)u(l) \tag{9}
\]
where the Markov parameters and the state transition matrix are defined for (1)–(2) by $h(k, k) = D(k)$, $\Phi(k, k) = I_n$ and
\[
\begin{align*}
h(k, l) &= C(k)\Phi(k, l+1)B(l), \quad l < k \\
\Phi(k, l) &= A(k-1) \cdots A(l).
\end{align*} \tag{10}
\]
Consider $h(k, l)$ terms in (9). If $k_l \leq l \leq k - 2$, from (10)
\[
\begin{align*}
h(k, l) &= C(k)A(k-1) \cdots A(l+1)B(l) \\
&= C_j A_{j_2}^{k-l-1}B_{j_2} = \hat{C}_j A_{j_2}^{k-l-1}B_{j_2}
\end{align*} \tag{11}
\]
which is still true if $k_l \leq l < k$. Note also that $h(k, k) = \hat{D}_{j_2}$ when $k \geq k_l$. Now, if $l \leq k_l - 2$ and $k > k_l$, from (3) and (10)
\[
\begin{align*}
h(k, l) &= C(k)A(k-1) \cdots A(k_l)A(k_l-1) \cdots A(l+1)B(l) \\
&= C_j A_{j_2}^{k-l-k_l}A_{j_1}^{k_l-l-1}B_{j_1} = \hat{C}_j A_{j_2}^{k-l-k_l}T_{j_2}T_{j_1}^{-1} A_{j_1}^{k_l-l-1}\hat{B}_{j_1},
\end{align*} \tag{12}
\]
valid also if $l < k_l$ which extends to $k \geq k_l$. Next, from $x(\xi_{i-1}) = T_{j_1}^{-1}\hat{x}(\xi_{i-1})$, (3), and (10) for $k \geq k_l$
\[
C(k)\Phi(k, \xi_{i-1})x(\xi_{i-1}) = C_j A_{j_2}^{k-l-k_l}A_{j_1}^{k_l-l-1}\hat{x}(\xi_{i-1})
\]
\[
C_j A_{j_2}^{k-l-k_l}T_{j_2}T_{j_1}^{-1} A_{j_1}^{k_l-l-1}\hat{B}_{j_1}x(\xi_{i-1}) = \hat{C}_j A_{j_2}^{k-l-k_l}T_{j_2}T_{j_1}^{-1} A_{j_1}^{k_l-l-1}\hat{x}(\xi_{i-1}). \tag{13}
\]
Hence, from (9), (11)–(13)
\[
z(k) = \hat{C}_j A_{j_2}^{k-l-k_l}T_{j_2}T_{j_1}^{-1} \kappa(\xi_{i-1}), \quad k \in [k_l, k_{i+1}) \tag{14}
\]
where
\[
z(k) = \begin{cases} y(k) - \hat{D}_{j_2}u(k) - \sum_{l=k_l}^{k-1} \hat{C}_j A_{j_2}^{k-l-l-1}B_{j_2}u(l), \quad k_i < k \\
y(k) - \hat{D}_{j_2}u(k), \quad k = k
\end{cases} \tag{15}
\]
\[
\kappa(\xi_{i-1}) = \hat{A}_{j_2}^{k_l-l-1}\hat{x}(\xi_{i-1}) + \sum_{l=\xi_{i-1}}^{k_l-1} \hat{A}_{j_1}^{k_l-l-1}\hat{B}_{j_1}u(l). \tag{16}
\]
From the input-output data and $\hat{D}_{j_2}$, $z(k)$ is computable on $[k_l, k_{i+1})$. In $\kappa(\xi_{i-1})$, $k$ does not appear. So does $\xi_{i-1}$.

**Proposition 3.1** Assume $\hat{A}_{j_2}$ is non-singular. Then $\kappa(\xi_{i-1})$ defined in (16) is constant on $[k_{i-1}, k_i - n]$.

**Proof.** Write (11) backward in time
\[
x(\xi_{i-1} - 1) = A_{j_1}^{-1}x(\xi_{i-1}) - A_{j_1}^{-1}B_{j_1}u(\xi_{i-1} - 1). \tag{17}
\]
Recall that \( \hat{x}(\xi_{i-1}) = T_{ji}x(\xi_{i-1}) \) for all \( k_{i-1} \leq \xi_{i-1} \leq k_i - n \). Then, from (17) if \( \xi_{i-1} > k_{i-1} \),

\[
\hat{x}(\xi_{i-1} - 1) = T_{ji}x(\xi_{i-1} - 1) = T_{ji}A_{ji}^{-1}x(\xi_{i-1}) - T_{ji}A_{ji}^{-1}B_{ji}u(\xi_{i-1} - 1) = \hat{A}_{ji}^{-1}T_{ji}x(\xi_{i-1}) - \hat{A}_{ji}^{-1}B_{ji}u(\xi_{i-1} - 1) = \hat{A}_{ji}^{-1}\hat{x}(\xi_{i-1}) - \hat{A}_{ji}^{-1}\hat{B}_{ji}u(\xi_{i-1} - 1).
\]

Plug \( \xi_{i-1} - 1 \) in (16) to get from the last equation above

\[
\kappa(\xi_{i-1} - 1) = \hat{A}_{ji}^{-1}\hat{x}(\xi_{i-1} - 1) + \sum_{l=\xi_{i-1} - 1}^{k_{i-1} - 1} \hat{A}_{ji}^{l-1}\hat{B}_{ji}u(l) = \hat{A}_{ji}^{-1}\hat{x}(\xi_{i-1}) - \hat{A}_{ji}^{-1}\hat{B}_{ji}u(\xi_{i-1} - 1) + \sum_{l=\xi_{i-1}}^{k_{i-1} - 1} \hat{A}_{ji}^{l-1}\hat{B}_{ji}u(j) - \hat{A}_{ji}^{-1}\hat{B}_{ji}u(\xi_{i-1} - 1) = \kappa(\xi_{i-1}).
\]

The iterations stop at \( \xi_{i-1} = k_{i-1} \).

Proposition 3.1 shows that further information cannot be extracted by considering initial state estimates at more than one point in \( [k_{i-1}, k_i - 1] \). To calculate \( \kappa(\xi_{i-1}) \), the inputs on \( [\xi_{i-1}, k_{i-1} - 1] \) are used while \( \hat{x}(\xi_{i-1}) \) is calculated from the input-output data on \( [\bar{\xi}_{i-1}, \eta_{i-1}] \). If \( q_{i-1} = n - 1 \), the \( n - 1 \) measurements coincide. An alternative is to use non-overlapping data blocks and extrapolate a past initial state to estimate \( \hat{x}(\xi_{i-1}) \) using (1) and \( \hat{x}(\xi_{i-1}) = T_{ji}x(\xi_{i-1}) \). Both approaches yield identical state estimates \( \hat{x}(\xi_{i-1}) \).

Let \( Y(j_1, j_2) = T_{ji}T_{ji}^{-1} \). We will show that if \( u(k) \) is persistently exciting (PE) and \( (j_1, j_2) \) is observed at least \( n \) times in the switching sequence, \( Y(j_1, j_2) \) satisfying (14) could be determined uniquely. To show this, fix \( q_i = n - 1 \) and \( \bar{\xi}_i = k_i \) for \( i > 0 \) and let \( Z(k) = [\zeta^T(k) \cdots \zeta^T(k + q_i)]^T \). From (14),

\[
Z(k_i) = \hat{\varrho}_{j_2}Y(j_1, j_2)\kappa(k_{i-1}).
\]

Since \( \hat{\varrho}(j_2) = \varrho(j_2)T_{ji}^{-1} \) and \( \varrho(j_2) \) has full rank,

\[
Y(j_1, j_2)\kappa(k_{i-1}) = \hat{\varrho}(j_2)Z(k_i).
\]

Let \( \text{card}(X) \) denote the cardinality of a given set \( X \) and

\[
\chi(j_1, j_2) = \{k_i : \varphi(k_{i-1}) = j_1, \varphi(k_i) = j_2\}.
\]

Assuming \( \text{card}(\chi(j_1, j_2)) \geq n \), stack (19) column-wise

\[
Y(j_1, j_2)\kappa(\chi(j_1, j_2) - 1) = \hat{\varrho}(j_2)Z(\chi(j_1, j_2))
\]

where \( \kappa(X) = \{\kappa(x) : x \in X\} \). If \( \kappa(\chi(j_1, j_2) - 1) \) has full rank, (21) has a unique solution. If the inputs are sufficiently rich, then \( \text{card}(\chi(j_1, j_2)) \geq n \) and the full-rank assumptions will easily be met as \( N \) grows.

From the definition of the dwell times,

\[
\kappa(k_i - 1) = \hat{A}_{ji}^\delta_{i-1}(X)\hat{x}(k_{i-1}) + \sum_{\varepsilon = 0}^{\delta_i - 1(\chi) - 1} \hat{A}_{ji}^\varepsilon\hat{B}_{ji}w(\varepsilon)
\]

where the change of the variables \( w(\varepsilon) = u(k_i - \varepsilon - 1) \) was made. As \( \delta_i \rightarrow \infty \), \( \hat{A}_{ji}^\delta_{i-1}(\chi)\hat{x}(k_{i-1}) \rightarrow 0 \) rapidly and \( \kappa(k_{i-1}) \approx \sum_{\varepsilon = 0}^{\delta_i} \hat{A}_{ji}^\varepsilon\hat{B}_{ji}w(\varepsilon) \) is a good approximation if \( u(k) \) is a quasi-stationary input.
sequence and $\kappa(k_i - 1)$ has a positive-definite covariance since $(\hat{A}_{j_i}, \hat{B}_{j_i})$ is controllable. The columns of $\kappa(\chi^{-1}(j_1, j_2))$ are built from the input measurements on non-overlapping segments, thus they are independent if $u(k)$ is a white-noise sequence. It follows that $\text{rank}(\kappa(\chi(j_1, j_2))) = n$ if $u(k)$ is a white-
noise sequence where $\text{rank}(X)$ denotes the rank of a given matrix $X$. The rank guarantee more generally follows from the following.

**Assumption 3.1** The inputs are PE, that is,

$$\text{rank}(\kappa(X)) = n$$

for all $n$-point sets $X = \{k_i, \cdots, k_n\} \subset \chi$.

One question remains to be settled if it is possible to relax the assumption $\text{card}(\chi(j_1, j_2)) \geq n$ by picking more points from $[k_i, k_{i+1})$. The answer is negative as demonstrated next. This means that from each of the segments $[k_{i-1}, k_i)$ and $[k_i, k_{i+1})$ one point should be selected and this process must be repeated at least $n$-times to determine $\Upsilon(j_1, j_2)$ uniquely from (21) when Assumption 3.1 holds. In fact, from (14)

$$\Upsilon(j_1, j_2)\kappa(k_{i-1}) = \hat{\Lambda}_{j_2}^{(k-k_i)} \hat{\Phi}_{j_2}^T Z(k)$$

for all $k \in [k_i, k_{i+1} - n]$ which is nothing, but (19).

If $\text{card}(\chi(j_1, j_2)) < n$, then (21) does not have a unique solution and a solution consistent with (21) may not satisfy (19) for a switch $k_i$ with $k_i > N, \varphi(k_{i-1}) = j_1$, and $\varphi(k_{i}) = j_2$. If $\text{card}(\chi(j_1, j_2)) \geq n$, (19) is always un-falsified with all switches after $N$ satisfying $\varphi(k_{i-1}) = j_1$ and $\varphi(k_i) = j_2$. The uniqueness issue has also an important consequence. In the sequel, we will show that it is not necessary to consider all pairs $\Upsilon(j_1, j_2), j_1, j_2 \in \mathcal{S}$ satisfying (19), but only a subset $\Upsilon(j, j + 1), 0 \leq j < \sigma$. The rest of the pairs are generated from this set by matrix inversions and products. Hence, the uniqueness property propagates to the entire set of all pairs. A direct consequence is that $N$ and $\text{card}(\chi)$ do not have to be excessively large to carry out a basis transformation.

### 3.3 A basis transformation algorithm

The discrete states, returned for example by an identification algorithm, may be in equivalence classes differing only by similarity transformations in the same class. They first need to be put into clusters by a statistics. We will use the $\ell_1$-norm of the eigenvalues $\hat{A}_\varphi(k)$ as a statistics defined by

$$\mathcal{M}(\hat{A}_\varphi(k)) = \sum_{\mu=1}^{n} |\lambda_\mu(\hat{A}_\varphi(k))|$$

provided that $\mathcal{M}(\hat{A}_\varphi(k)) = \mathcal{M}(\hat{A}_\varphi(l))$ iff $\varphi(k) = \varphi(l)$. By viewing the graph of $\mathcal{M}(\hat{A}_\varphi(k)), 1 \leq k \leq N, \sigma$ clusters of the discrete states in $\mathcal{P}$ are determined since every discrete state in $\mathcal{P}$ is active in at least one segment. We assume that $\varphi$ satisfies a saturation condition on $[1, N]$ as follows.

**Assumption 3.2** Let $\chi(j_1, j_2)$ be as in (20). Then, for all $0 \leq j < \sigma$, $\text{card}(\chi(j, j + 1)) \geq n$.

The collection $\chi(j, j + 1), j = 0, \cdots, \sigma - 1$ in Assumption 3.2 is minimal in the sense that if we start on a segment in which $\mathcal{P}_\sigma$ is active, then a segment in which $\mathcal{P}_1$ is active is reachable by passing through several switches on a chosen path. Let $(\sigma, j_1, j_2, \cdots, j_k, 1)$ be such a path allowing returns to the same discrete states. The chain of equalities

$$\Upsilon^{-1}(\sigma, j_1) \Upsilon^{-1}(j_1, j_2) \cdots \Upsilon^{-1}(j_k, 1) = T_\sigma T_{j_1}^{-1} T_{j_2}^{-1} \cdots T_{j_k}^{-1} T_1^{-1} = T_\sigma T_1^{-1} = Y_1, \sigma = Y_{\sigma, 1}$$

6
shows that only the initial and the final discrete states matter. Moreover, since \( \mathcal{Y}_{j_1,j_2} = T_{j_2}^{-1} \mathcal{Y}^{-1}_{j_1} \), it is sufficient to calculate either \( \mathcal{Y}_{j_1,j_2} \) or \( \mathcal{Y}^{-1}_{j_2,j_1} \). By using chains and inversions as often as needed, any \( \mathcal{Y}(j_1,j_2) \) can be written as a product of the factors \( \mathcal{Y}^{\pm 1}(j,j+1) \). Under Assumptions 3.1 and 3.2 note that \( \mathcal{Y}(j_1,j_2) \) is unique for all \( j_1, j_2 \in \mathbb{S} \).

We define a basis transform by \( \tilde{\mathcal{P}}_j = (\tilde{A}_j, \tilde{B}_j, \tilde{C}_j, \tilde{D}_j) \) where \( \tilde{A}_j = \mathcal{Y}^{-1}_{1,j} \hat{A}_j \mathcal{Y}_{1,j} \), \( \tilde{B}_j = \mathcal{Y}^{-1}_{1,j} \hat{B}_j \), \( \tilde{C}_j = \hat{C}_j \mathcal{Y}_{1,j} \), \( \tilde{D}_j = \hat{D}_j \) and calculate the factors of \( z(k) \) for \( k \in [k_i,k_{i+1}] \) starting with

\[
\tilde{C}_j \hat{A}_j^{k-i} T_{j_i} T_{j_i}^{-1} = \hat{C}_j \mathcal{Y}^{-1}_{1,j} \mathcal{Y}_{1,j} \hat{A}_j^{k-i} \mathcal{Y}^{-1}_{1,j} T_{j_i} T_{j_i}^{-1} = \hat{C}_j \tilde{A}_j^{k-i} \mathcal{Y}(j_i,1) .
\]

Denote the extended observability matrix of \( \tilde{\mathcal{P}}_j \) by \( \tilde{O}(j_i) \). From \( \mathcal{O}(j_i) = \mathcal{O}(j_i) \mathcal{Y}^{-1}(1, j_i) \) and (8),

\[
\hat{x}(k_{i-1}) = \mathcal{O}^T(j_i) (\mathcal{Y}_{i-1} - \Gamma(j_i) U_{i-1}) = \mathcal{Y}(1, j_i) \mathcal{O}^T(j_i) (\mathcal{Y}_{i-1} - \Gamma(j_i) U_{i-1}) = \mathcal{Y}(1, j_i) \hat{x}(k_{i-1})
\]

and hence

\[
\kappa(k_{i-1}) = \hat{A}_j^{k_i-i} \hat{x}(k_{i-1}) + \sum_{l=k_i}^{k_{i-1}} \hat{A}_j^{k_i-l} \hat{B}_j u(l) = \mathcal{Y}(1, j_i) \left[ \hat{A}_j^{k_i-i} \hat{x}(k_{i-1}) + \sum_{l=k_i}^{k_{i-1}} \hat{A}_j^{k_i-l} \hat{B}_j u(l) \right] = \mathcal{Y}^{-1}(j_i,1) \hat{x}(k_{i-1}) .
\]

It follows that \( z(k) = \tilde{C}_j \tilde{A}_j^{k-i} \hat{x}(k_{i-1}) \) for all \( k \in [k_i,k_{i+1}] \). Since \( \tilde{\mathcal{P}}_{j_2} \sim \mathcal{P}_{j_2} \) on this interval, their Markov parameters are matched there and from (15) we see that this representation is valid also for \( y(k) \) without any dependence on the similarity transformations. A pseudo-code implementing the results derived above is outlined in Algorithm 1. The result derived in this paper compares favorably to that in [19]. Due to the utilization of chains and inversions fewer switches and similarity transformations are involved in our calculations.

**Algorithm 1. Basis transform for discrete states**

**Inputs:** \( u(k), y(k), \tilde{\mathcal{P}}_k, \chi(k), \) for all \( k \in [1,N], \)

1. Decompose \( \tilde{\mathcal{P}}_k, k \in [1,N] \) into \( \sigma \)-clusters via (25)
2. For \( 0 \leq j < \sigma \) and \( j_1 = j_i + 1 \), solve (21).
3. Use the chain rule to calculate \( \mathcal{Y}(1,j), j \geq 3 \)
4. Calculate \( \tilde{\mathcal{P}}_j, j = 1, \cdots, \sigma \).

**Outputs:** \( \tilde{\mathcal{P}}_j, j = 1, \cdots, \sigma \).

We state the result derived in this paper as follows.

**Theorem 3.1** If \( \mathcal{P}_j, 1 \leq j \leq \sigma \) have no poles at zero, \( \delta_i(\chi) \geq n \) for all \( 0 \leq i < i' \), and Assumptions 3.1–3.2 hold, then Algorithm 1 solves Problem 2

### 3.4 Relations with the graph theory

Recall that there are \( \sigma(\sigma-1)/2 \) pairs \( (j_1,j_2) \in \mathbb{S}^2 \) with distinct indices \( j_1 \neq j_2 \). If \( (j_1,j_2) \) is not distinguishable from \( (j_2,j_1) \), the number of the pairs is \( \sigma(\sigma-1)/2 \). Let us consider those pairs with \( \text{card}(\chi(j_1,j_2)) \geq n \). From (21), we then calculate \( \mathcal{Y}(j_1,j_2) \) uniquely, hence by inversion \( \mathcal{Y}(j_2,j_1) \) uniquely as well, meaning that the order of appearance is irrelevant as long as \( j_1 \neq j_2 \) and \( \text{card}(\chi(j_1,j_2)) \geq n \).
n. In this subsection, we shall answer the question “What is the minimum number of the pairs that must be learned to be able to extract the rest of the pairs?” Obviously, this is a relevant question to basis construction since asymptotically every pair satisfies $\text{card}(\chi(j_1, j_2)) \geq n$. To answer this question, we represent the underlying setup via an undirected graph where the vertices (nodes) are the discrete states. For the sake of simplicity, we consider the $\sigma = 4$ case; the $\sigma > 4$ cases follow similarly. Omitting reverse edges because they are bidirectional, in the example we are given the vertex and the edge sets by

$$V = \{1, 2, 3, 4\},$$
$$E = \{(1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)\}.$$

A graph representation for $V$ and $E$ is shown in Figure 1.

![Figure 1: A graph representation for $V$ and $E$.](image)

Explanation for the graph representation is as follows. Existence of an edge $(j_1, j_2)$ between two nodes $j_1$ and $j_2$ means that $\Upsilon(j_1, j_2)$ is determined uniquely. The graph in Figure 1 is complete if there is an edge between any two vertices. A graph with this property has $\sigma(\sigma - 1)/2$ edges. The goal here is to reduce the number of the edges to a minimum while keeping the graph connected. Two vertices $j_1$ and $j_2$ are connected if $G$ contains a path from $j_1$ to $j_2$. A graph is said to be connected if every pair of vertices in the graph is connected. We reduce edges in a graph until all edges are of bridge type. A bridge is an edge if removing it from a connected graph, renders the graph disconnected. After all reductions, the graph is said a spanning tree which we denote by $T$. Trees are by construction connected graphs with only one path joining any two vertices. It becomes a spanning tree of $G$ when it contains all the vertices originally found in $G$. We can always find spanning trees for connected graphs. If a graph has cycles, we only need to remove enough edges to ensure that the reduced graph is connected and no cycles remain [20]. Any spanning tree $T$ of $G$ has $\sigma$ vertices. It is a basic fact that a tree with $\sigma$ vertices has $\sigma - 1$ edges. Therefore, to obtain a spanning tree, we must adhere the rules:

1. All vertices from $G$ must be present.
2. There must be exactly $\sigma - 1$ edges, all nodes connected and each node connected to at least one edge.

From the Cayley’s formula, number of spanning trees for a complete graph with $\sigma$ vertices is known as $\sigma^{\sigma-2}$. Thus, the graph in Figure 1 admits 16 spanning trees with 3 edges each. We deduce that $\sigma - 1$ matrices $\Upsilon(j_1, j_2)$ are required with $j_1$ and $j_2$ surjective to $S$ and there are $\sigma^{\sigma-2}$ collections of $\sigma - 1$ matrices. Two spanning trees are shown in Figure 2a–2b.
We apply the graph theory to the basis construction problem for an SLS with 4 discrete states. Disregarding order of appearances, there are 6 edges $\Gamma(j_1, j_2)$. List them all:

$$\Gamma^* = \{\Gamma(1, 2), \Gamma(1, 3), \Gamma(1, 4), \Gamma(2, 3), \Gamma(2, 4), \Gamma(3, 4)\}.$$  

The graph representation is shown in Figure 1. The tree in Figure 2a is the edge subset $\Gamma^1 = \{\Gamma(1, 2), \Gamma(1, 3), \Gamma(1, 4)\}$ with the removed edges calculated as follows

$$\Gamma(2, 3) = \Gamma(1, 3)\Gamma^{-1}(1, 2),$$
$$\Gamma(2, 4) = \Gamma(1, 4)\Gamma^{-1}(1, 2),$$
$$\Gamma(3, 4) = \Gamma(1, 4)\Gamma^{-1}(1, 3).$$

Similar conclusions are drawn for the spanning tree in Figure 2b. However, if we select $\Gamma^2 = \{\Gamma(1, 2), \Gamma(1, 4), \Gamma(2, 4)\}$, the removed edges are not retrievable although $\Gamma^2$ has 3 elements. In Figure 3, $\Gamma^2$ and the vertex set are plotted. This spanning subgraph is not a spanning tree since it violates the second rule. It is disconnected since Node 3 can not be reached from any other nodes. The edges $\Gamma(j, j + 1)$, $j = 0, \cdots, \sigma - 1$ in Assumption 3.2 forms a spanning tree.

4  Numerical example

To illustrate the results derived in this paper, we consider an SLS model with three discrete states in the state-space form
number of 14 segments. Note that detection techniques through a numerical study.

The bimodal SLS formed by the first two discrete states was used in [12] to investigate performance of a hybrid identification algorithm that merges Markov parameter based subspace identification and change detection techniques through a numerical study.

A switching sequence that adheres to the dwell time constraint \( \min \delta_i(\chi) \geq n \) and conforms with Assumption 3.2 was sampled from a random uniform distribution with its plot given in Figure 4. The collection \( \hat{\mathcal{P}}_{\varphi(k)} \), \( 1 \leq k \leq N \) was generated from \( \mathcal{P}_{\varphi(k)} \), \( 1 \leq k \leq N \) by applying randomly generated linear transformations, that is, sole linear transformations applied to the discrete states residing on the segments. This set-up imitates a frequently encountered scenario in the scope of SLS identification [12, 21, 22]. The switching sequence in Figure 4 is characterized by the path \{2, 3, 2, 3, 1, 2, 1, 3, 2, 3, 1, 2, 1, 2\} with a total number of 14 segments. Note that

\[
\chi(1, 2) = \{k_5, k_{11}, k_{13}\}, \quad \chi(2, 3) = \{k_1, k_3, k_9\},
\]

and hence \( \text{card}(\chi(1, 2)) = \text{card}(\chi(2, 3)) = 3 = n \).

For each segment \([k_i, k_{i+1})\), we randomly generated a linear transformation \( T_i \) and compute \( \hat{\mathcal{P}}_{\varphi(k)} \) according to

\[
\hat{A}_k = T_i A_{\varphi(k)} T_i^{-1}, \quad \hat{B}_k = T_i B_{\varphi(k)}, \quad \hat{C}_k = C_{\varphi(k)} T_i^{-1}, \quad \hat{D}_k = D_{\varphi(k)},
\]

\[
A_1 = \begin{pmatrix} 0.32 & 0.31 & 0 \\ -0.32 & 0.31 & 0 \\ 0 & 0 & -0.18 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 0.90 & -0.70 \\ 0.71 & -0.50 \\ 0.80 & 0.47 \end{pmatrix},
\]

\[
C_1 = \begin{pmatrix} -0.55 & 0.20 & 0.80 \\ 0.45 & 0.30 & 0.58 \end{pmatrix}, \quad D_1 = \begin{pmatrix} 0.90 & -0.70 \\ 0.71 & -0.50 \\ 0.80 & 0.47 \end{pmatrix};
\]

\[
A_2 = \begin{pmatrix} -0.10 & -0.40 & 0 \\ 0.50 & -0.40 & 0 \\ 0 & 0 & 0.26 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0.10 & -0.60 \\ 0.32 & -0.66 \\ 0.30 & 0.82 \end{pmatrix},
\]

\[
C_2 = \begin{pmatrix} -0.80 & -0.10 & 0.70 \\ 0.30 & 0.48 & 0.90 \end{pmatrix}, \quad D_2 = \begin{pmatrix} 0.50 & 0.30 \\ -0.20 & -0.50 \end{pmatrix};
\]

\[
A_3 = \begin{pmatrix} 0.4 & 0.1 & 0 \\ 0.8 & 0.4 & 0 \\ 0 & 0 & 0.8 \end{pmatrix}, \quad B_3 = \begin{pmatrix} 1.5 & 0.9 \\ 1 & -1 \\ -1.5 & 2.3 \end{pmatrix},
\]

\[
C_3 = \begin{pmatrix} 0.8 & 1.1 & 2 \\ -1.3 & 0.7 & 1.7 \end{pmatrix}, \quad D_3 = \begin{pmatrix} 1 & 0 \\ 1 & 2 \end{pmatrix}.
\]

The bimodal SLS formed by the first two discrete states was used in [12] to investigate performance of a hybrid identification algorithm that merges Markov parameter based subspace identification and change detection techniques through a numerical study.

Figure 3: A graph representation for \( Y^2 \) and the vertex set.

![Graph Representation](image-url)
for all $k \in [k_i, k_{i+1})$. We collected input-output data $u(k), y(k), k \in [1, N]$ by feeding superposed harmonic input signal satisfying Assumption 3.1 through the actual SLS model. The same data set was used to drive Algorithm 1.

A cluster set $C$ was identified by executing the `dbscan` command in MATLAB [23] with $M(\hat{A}(k))$ for $k \in [1, N]$ and selecting $\text{epsilon}=10^{-4}$ as the threshold for a neighborhood search radius and $\text{minpts}=20$ for the minimum number of neighbors. The graph of $M(\hat{A}(k))$ in Figure 5a reveals three different levels discerned easily. Figure 5b, likewise, outputs the histogram of clustering. As anticipated, $\sigma$ is correctly estimated. If the discrete state residing in the $j$th segment is $\hat{P}_j$, then

$C = C_1 \cup C_2 \cup C_3$. Each element $C_j, j = 1, 2, 3$ is an equivalence class ordered by similarity. We pick a representative from each equivalence class. Such a representative could be chosen randomly from the class. To keep it simple, we choose the first element of each class as the representative. We end up retrieving the $\hat{\sigma}$ identified discrete state set. $\{\hat{P}_1, \hat{P}_2, \hat{P}_3\} = \{\hat{P}_5, \hat{P}_1, \hat{P}_2\}$. We list the state-transition matrices of the identified submodels:

$$\hat{A}_1 = \begin{pmatrix} -0.22 & 0.87 & -0.51 \\ -0.03 & 0.67 & -0.51 \\ -0.99 & 1.39 & 0 \end{pmatrix},$$

$$\hat{A}_2 = \begin{pmatrix} 1.31 & -3.81 & 1.51 \\ 1.06 & -3.61 & 1.81 \\ 1.09 & -3.99 & 2.06 \end{pmatrix},$$

$$\hat{A}_3 = \begin{pmatrix} 0.9 & 0.04 & -0.10 \\ -1.08 & 0.05 & 1.24 \\ 0.15 & 0 & 0.65 \end{pmatrix}.$$
We applied $\Upsilon(2,1)$ and $\Upsilon(3,1)$ to $\mathcal{P}_2$ and $\mathcal{P}_3$, respectively, yielding $\mathcal{P}_2$ and $\mathcal{P}_3$ while $\mathcal{P}_1 = \mathcal{P}_1$. 

**Figure 5:** Retrieval of the clusters via the `dbscan` command.

---

**References**

1. M. $\hat{A}(k)$
2. $\Upsilon(1,3)$ from the chain rule $\Upsilon(1,3) = \Upsilon(2,3)\Upsilon(1,2)$. Thus, we get

$$
\Upsilon(1,2) = \begin{pmatrix}
0.47 & 0.21 & 0.72 \\
-0.57 & 1.45 & 0.1 \\
-1.67 & 2.53 & 0.26 \\
\end{pmatrix}
$$

$$
\Upsilon(1,3) = \begin{pmatrix}
2.60 & -2.69 & 0.21 \\
0.05 & 0.95 & -0.28 \\
1.18 & -1.30 & 0.92 \\
\end{pmatrix}.
$$
This operation brings the state basis of $\mathcal{P}_2$ and $\mathcal{P}_3$ to that of $\mathcal{P}_1$ permitting matching of the Markov parameters. The success of the presented approach is checked by monitoring the mismatch error between the predicted outputs and the actual ones to the superposed harmonic inputs. Figure 6 displays the true output signals and the estimation errors side by side. The match between the two is perfect.

![Graphs of outputs and errors](image-url)
5 Conclusions

In this paper, we addressed the basis transformation issue for switched linear systems from input-output data and the locally identified submodels. We provided an algorithm to compute a transformation from the input-output data and a given collection of the local models that leaves the input-output maps invariant. The efficacy of the proposed approach was demonstrated through a numerical example. A link between the basis transformation problem for the SLSs and the graph theory was put forward.

References

[1] B. Lennartson, M. Tittus, B. Egardt, and S. Pettersson, “Hybrid systems in process control,” *IEEE Control Systems Magazine*, vol. 16, pp. 45–56, 1996.

[2] R. Carloni, R. G. Sanfelice, A. R. Teel, and C. Melchior, “A hybrid control strategy for robust contact detection and force regulation,” in: Proc. *American Control Conf.*, New York, pages 1461–1466, July 2007.

[3] T. Schlegl, M. Buss, and G. Schmidt, “A hybrid systems approach toward modeling and dynamical simulation of dexterous manipulation,” *IEEE/ASME Trans. Mech.*, vol. 8, pp. 352–361, 2003.

[4] X. Jin and B. Huang, “Robust identification of piecewise/switching autoregressive exogenous process,” *AIChE Journal*, vol. 56, pp. 1829–1844, 2010.

[5] W. Glover and J. Lygeros, “A stochastic hybrid model for air traffic control simulation,” in: Proc. *Int. Workshop Hybrid Systems: Comput. Control*, Philadelphia, PA, pages 372–386, March, 2004.

[6] M. Prandini and J. Hu, “Application of reachability analysis for stochastic hybrid systems to aircraft conflict prediction,” in: Proc. *47th IEEE Conf: Decision Contr.*, pages 4036–4041, December 2008.

[7] J. Lee, S. Bohacek, J. P. Hespanha, and K. Obrazcza, “Modeling communication networks with hybrid systems,” *IEEE/ACM Transactions on Networking*, vol. 15, pp. 630–643, 2007.

[8] A. Germani, C. Manes, and P. Palumbo, “Simultaneous system identification and channel estimation: a hybrid system approach,” in: Proc. *46th IEEE Conference on Decision and Control*, New Orleans, LA, pages 1764–1769, December 2007.

[9] A. Cetinkaya, H. Ishii, and T. Hayakawa, “Analysis of stochastic switched systems with application to networked control under jamming attacks,” *IEEE Transactions on Automatic Control*, vol. 64, pp. 2013–2028, 2018.

[10] M. S. Mahmoud and Y. Xia, *Cloud Control Systems: Analysis, Design and Estimation*, Academic Press: London, , 2020.

[11] I. A. Hiskens and M. Pai, “Hybrid systems view of power system modelling,” in: Proc. *IEEE Int. Symp. Circuits and Systems*, Geneva, Switzerland, pages 228–231, May 2000.

[12] K. M. Pekpe, G. Mourot, K. Gasso, and J. Ragot, “Identification of switching systems using change detection technique in the subspace framework,” in: Proc. *43rd IEEE Conference on Decision and Control*, Nassau, Bahamas, pages 3720–3725, December 2004.

[13] R. Murray-Smith, “Modelling human control behaviour with context-dependent Markov-switching multiple models,” *IFAC Proceedings Volumes*, vol. 31, pages 461–466, 1998.

[14] R. Vidal and Y. Ma, “A unified algebraic approach to 2-d and 3-d motion segmentation and estimation,” *Journal of Mathematical Imaging and Vision*, vol. 25, pp. 403–421, 2006.
[15] E. Cinquemani, A. Milias-Argeitis, and J. Lygeros, “Identification of genetic regulatory networks: A stochastic hybrid approach,” IFAC Proceedings Volumes, vol. 41, pages 301–306, 2008.

[16] V. Verdult and M. Verhaegen, “Subspace identification of piecewise linear systems,” in: Proc. 43rd IEEE Conference on Decision and Control, Nassau, Bahamas, pages 3838–3843, December 2004.

[17] T. Kailath, Linear systems, Prentice-Hall: Englewood Cliffs, NJ, 1980.

[18] G. Mercère and L. Bako, “Parameterization and identification of multivariable state-space systems: A canonical approach,” Automatica, vol. 47, pp. 1547–1555, 2011.

[19] J. Borges, V. Verdult, and M. Verhaegen, “Iterative subspace identification of piecewise linear systems,” in: Proc. 14th IFAC Symp. Syst. Ident., Newcastle, Australia, pages 368–373, March 2006.

[20] K. R. Saoub, Graph Theory: An Introduction to Proofs, Algorithms, and Applications, Boca Raton: CRC Press, 2021.

[21] R. V. Lopes, J. Y. Ishihara, and G. A. Borges, “Identification of state-space switched linear systems using clustering and hybrid filtering,” Journal of the Brazilian Society of Mechanical Sciences and Engineering, vol. 39, pp. 565–573, 2017.

[22] F. Bencherki, S. Türkay, and H. Akçay, “State-space identification of switched linear systems by sparse optimization,” Preprint.

[23] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in: Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining, Portland, OR, pages 226–231, 1996.