Consistent and computationally efficient estimation for stochastic LPV state-space models: realization based approach

Manas Mejari, Mihaly Petreczky

Consistent and computationally efficient estimation for stochastic LPV state-space models: realization based approach.

Manas Mejari and Mihály Petreczky

***Abstract*— The article presents an identification algorithm for stochastic *Linear Parameter-Varying State-Space Affine* (LPV-SSA) representations, where the dependency of state-space matrices on scheduling signals is affine. Based on stochastic realization theory, a computationally efficient and statistically consistent identification algorithm is proposed to estimate the LPV model matrices, which are computed from the empirical covariance matrices of outputs and scheduling signal observations. The effectiveness of the proposed realization algorithm is shown via a numerical case study.**

I. INTRODUCTION

The *Linear Parameter-Varying* (LPV) modeling paradigm can be viewed as a natural extension of *Linear Time-Invariant (LTI)* model class, having linear dynamic relation between input and output signals. However, unlike LTI systems, this relation can change over time according to a measurable time-varying signal, the so called *scheduling variable*. By the virtue of scheduling variables, non-linear and time-varying dynamics can be embedded via scheduling variable dependent LPV model coefficients. In this way, LPV models can describe the behavior of many time-varying and non-linear systems such as automobiles [4], [13], aircrafts [11], and distillation columns [1] etc. Naturally, identification and control of LPV models has attracted significant attention.

Over the past few years, many methods have been proposed for the identification of LPV models, in input-output (IO) [2], [12], [17] as well as state-space (SS) representations [9], [19], [21], [22], [20]. From a control perspective, the IO models are not well suited for controller synthesis. The controller design approaches (e.g., [18], [23]) often require the LPV models to be in SS representation with an affine dependency on the scheduling signal. To this end, *realization theory* of LPV models plays a key role. Realization theory aims at understanding the conditions under which the observed behavior of a system can be realized by a state-space representation. It also allows to formulate identification algorithms for estimating (preferably minimal) state-space representation from a finite set of observations. In fact, this paper formulates such an algorithm.

M. Mejari is with IDSIA Dalle Molle Institute for Artificial Intelligence SUPSI, 6928 Manno, Switzerland manas.mejari@idsia.ch

M. Petreczky is with the Centre de Recherche en Informatique, Signal et Automatique de Lille, UMR CNRS 9189, Ecole Centrale de Lille, Villeneuve dAscq 59651, France : mihaly.petreczky@ec-lille.fr

This work was partially funded by CPER Data project, which is co-financed by European Union with the financial support of European Regional Development Fund (ERDF), French State and the French Region of Hauts-de-France and by CNRS project PEPS blanc BayesRealForRNN.

The realization theory for deterministic *Linear Parameter-Varying State-Space Affine* (LPV-SSA) representation has been developed in [14]. The results of [14] have been used for model reduction in [3] and to derive LPV-SS identification algorithm in [6]. Recently, a maximum likelihood based LPV-SS identification algorithm has been proposed in [5] which combines LPV-IO identification methods with an LPV-IO to LPV-SS realization scheme. These methods are focused mainly on *deterministic* realizations, which for certain control and filtering problems are too restrictive. This shortfall can be overcome by modeling the behavior of the system using *stochastic* state-space representations.

Intuitively, in a stochastic setting, the LPV-SSA representation can be decomposed into *deterministic* and *stochastic* parts. Hence, intuitively, identification of stochastic LPV-SSA representations can be decomposed into two independent problems: identification of the deterministic part, and identification of the stochastic part. As it was pointed out above, there are provenly correct algorithms for identifying the deterministic part [5]. In this paper, we will focus on the identification of the stochastic part, i.e., on the identification of stochastic LPV-SSA representations *without inputs*, which only depend on noise processes. The contribution of this paper can be seen as the first step towards identification of the more general class of stochastic LPV-SSA representation with external input processes. Note that the argument of the decomposition above is not a mathematically strict one. Making this argument strict calls for developing realization theory of stochastic LPV representations with inputs. The latter was highly non-trivial even for linear time-invariant systems [10], and we leave it for future research.

The identification algorithm presented in this paper is a non-parametric one and it is based on stochastic covariance realization algorithm for LPV-SSA representations [16]. The main idea is to estimate the unknown model matrices by applying a modification of the covariance realization algorithm [16] to empirical covariances computed from the observed dataset. Furthermore, in order to reduce the computational complexity, the idea of *selections* from [5] is adapted to the stochastic realization algorithm [16]: instead of using the full Hankel-matrix, a suitable sub-matrix of the full Hankel matrix is selected for computing the estimates of system matrices and noise covariances. The proposed identification algorithm is both provenly consistent and computationally efficient.

Subspace based identification of stochastic LPV-SSA representations has a rich literature, e.g., [21], [7], [8]. The algorithm proposed in this paper differs significantly from [21],

[7], [8]. In contrast to subspace identification algorithms, the proposed algorithm is provenly consistent and it does not require the observability assumption of [21], [7], [8]. However, this comes at a price, as the proposed algorithm is provenly consistent only for a very specific class of scheduling signals and stochastic LPV-SSA representations. The proposed algorithm avoids the curse of dimensionality, but this comes at the price of either using some prior knowledge on the system to determine the correct selection of the rows and columns of a Hankel-matrix or using an exhaustive search to find such a selection.

Outline: The terminology and notation used throughout the paper is introduced in Section **??**. The identification problem for stochastic LPV-SSA representations is formalized in Section II. A computationally efficient stochastic realization algorithm is formulated in Section III, based on which an identification algorithm is developed in Section IV. A numerical example is reported in Section V to asses the performance of the proposed algorithm. Finally, concluding remarks are given in Section VI.

II. PROBLEM FORMULATION

We consider the following, discrete-time Linear Parameter-Varying State-Space Affine (LPV-SSA) representation

$$\begin{aligned}\mathbf{x}(t+1) &= A(\boldsymbol{\mu}(t))\mathbf{x}(t) + K(\boldsymbol{\mu}(t))\mathbf{v}(t),\\ \bar{\mathbf{y}}(t) &= C\mathbf{x}(t) + D\mathbf{v}(t),\end{aligned} \tag{1a}$$

where $\mathbf{x}$, $\bar{\mathbf{y}}$ and $\mathbf{v}$ denote *state process*, *output process* and *noise process* taking values in $\mathbb{R}^n$, $\mathbb{R}^p$, and $\mathbb{R}^m$ respectively, $\boldsymbol{\mu}$ denotes the scheduling signal process taking values in $\mathbb{R}^d$, and at every time instance $t \in \mathbb{Z}$, $\boldsymbol{\mu}(t) = [1, \boldsymbol{\mu}_2(t), \cdots, \boldsymbol{\mu}_d(t)]^\top$. The matrices $A(\cdot)$ and $K(\cdot)$ are *affine* functions of the scheduling signal $\boldsymbol{\mu}(t)$ given by,

$$A(\boldsymbol{\mu}(t)) = A_1 + \sum_{i=2}^{d} A_i \boldsymbol{\mu}_i(t),\ K(\boldsymbol{\mu}(t)) = K_1 + \sum_{i=2}^{d} K_i \boldsymbol{\mu}_i(t). \tag{1b}$$

The matrices $A_i \in \mathbb{R}^{n\times n}$, $K_i \in \mathbb{R}^{n\times m}$, $\forall i = 1, \ldots, d$, $C \in \mathbb{R}^{p\times n}$ and $D \in \mathbb{R}^{p\times m}$ are real, constant matrices. We define the noise covariance matrices Q_i for each $\mathbb{R}$-valued scheduling signal component $\boldsymbol{\mu}_i$ as, $Q_i = \mathbb{E}\left[\mathbf{v}(t)\mathbf{v}^\top(t)\boldsymbol{\mu}_i^2(t)\right]$, for all $i = 1, \ldots, d$. The LPV-SSA representation (1) is said to be a *realization* of the observed stochastic process $\mathbf{y}$, if for all $t \in \mathbb{Z}$, $\mathbf{y}(t) = \bar{\mathbf{y}}(t)$.

The identification problem can be stated as follows:

Problem 1: Assume that $\bar{y} : \mathbb{Z} \to \mathbb{R}^p$ is a sample path of the process $\mathbf{y}$ and $\bar{\mu} : \mathbb{Z} \to \mathbb{R}^d$ is a sample path of the process $\boldsymbol{\mu}$. Given a dataset $\{\bar{y}(t), \bar{\mu}(t)\}_{t=1}^N$ consisting of samples of the output and scheduling process, compute from this dataset the estimates $\{\{\hat{A}_i, \hat{K}_i, \hat{Q}_i\}_{i=1}^d, \hat{C}, \hat{D}\}$, such that as $N \to \infty$, the estimated matrices $\{\{\hat{A}_i, \hat{K}_i, \hat{Q}_i\}_{i=1}^d, \hat{C}, \hat{D}\}$ converge (in a suitable sense) to the matrices $\{\{A_i, K_i, Q_i\}_{i=1}^d, C, D\}$ of a true system, i.e., the LPV-SSA of the form (1) with $Q_i = \mathbb{E}\left[\mathbf{v}(t)\mathbf{v}^\top(t)\boldsymbol{\mu}_i^2(t)\right]$, $i = 1, \ldots, d$, is a realization of output process $\mathbf{y}$. In particular, for a large enough N, the output $\hat{\mathbf{y}}$ obtained from the estimated matrices $\{\{\hat{A}_i, \hat{K}_i\}_{i=1}^d, \hat{C}, \hat{D}\}$, is close to the observed output $\mathbf{y}$ in a suitable sense.

A. Technical preliminaries

Let us define the set $\Sigma = \{1, \ldots, d\}$. The set Σ will be referred to as the *alphabet*. The elements of Σ are termed as *letters*. A *non empty word* over Σ is a finite sequence of letters, i.e., $w = \sigma_1\sigma_2\cdots\sigma_k$, where $0 < k \in \mathbb{Z}$, $\sigma_1, \ldots, \sigma_k \in \Sigma$. The set of *all* nonempty words is denoted by Σ^+. We denote an *empty word* by ϵ. Let $\Sigma^* = \epsilon \cup \Sigma^+$. The concatenation of two nonempty words $v = a_1a_2\cdots a_m$ and $w = b_1b_2\cdots b_n$ is defined as $vw = a_1\cdots a_m b_1\cdots b_n$ for some $m, n > 0$. Note that if $w = \epsilon$ or $v = \epsilon$, then $v\epsilon = v$ and $\epsilon w = w$, moreover, $\epsilon\epsilon = \epsilon$. The length of the word $w \in \Sigma^*$ is denoted by $|w|$, and $|\epsilon| = 0$. Eg: for alphabet $\Sigma = \{1, 2\}$, $\Sigma^* = \{\epsilon, 1, 2, 11, 12, 21, 22 \ldots\}$, for the word $w = 111 \in \Sigma^*$, $|w| = 3$.

In this paper, we consider zero-mean independent identically distributed (i.i.d.) scheduling process $\boldsymbol{\mu} = [1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_d]^\top$ such that for $\Sigma = \{1, \ldots, d\}$, we have $\boldsymbol{\mu}_1 = 1$, and for each $i = 2, \ldots, d$, $\boldsymbol{\mu}_i$ is zero mean i.i.d. process, and for all $t \in \mathbb{Z}$, we have $\mathbb{E}\left[\boldsymbol{\mu}_\sigma^2(t)\right] = p_\sigma$, $\forall \sigma \in \Sigma$. In particular, $p_1 = 1$.

Notations 1: For each word $w \in \Sigma^*$, we define the number p_w as follows: for an empty word $w = \epsilon$, we define $p_\epsilon = 1$, and if $w = v\sigma$, then $p_w = p_v p_\sigma$ for $v \in \Sigma^*$ and $\sigma \in \Sigma$.

For every word $w \in \Sigma^+$ where $w = \sigma_1\sigma_2\cdots\sigma_k$, $k \geq 1$, $\sigma_1, \ldots, \sigma_k \in \Sigma$, we define the input process $\boldsymbol{\mu}_w$ as

$$\boldsymbol{\mu}_w(t) = \boldsymbol{\mu}_{\sigma_1}(t-k+1)\boldsymbol{\mu}_{\sigma_2}(t-k+2)\cdots\boldsymbol{\mu}_{\sigma_k}(t), \forall t \in \mathbb{Z}. \tag{2}$$

For a stochastic process $\mathbf{y} \in \mathbb{R}^k$, we define for each $w \in \Sigma^+$, the process $\mathbf{z}_w^{\mathbf{y}}$ as follows

$$\mathbf{z}_w^{\mathbf{y}}(t) = \mathbf{y}(t - |w|)\boldsymbol{\mu}_w(t-1)\frac{1}{\sqrt{p_w}},\ \forall t \in \mathbb{Z}. \tag{3}$$

Notations 2 (Matrix Product): Consider a collection of square matrices $A_\sigma \in \mathbb{R}^{n\times n}$, $\sigma \in \Sigma$. For any word $w \in \Sigma^+$ of the form $w = \sigma_1\sigma_2\cdots\sigma_k$, $k > 0$ and $\sigma_1, \ldots, \sigma_k \in \Sigma$, we define the following matrix product notation: $A_w = A_{\sigma_k}A_{\sigma_{k-1}}\cdots A_{\sigma_1}$. For an empty word ϵ, $A_\epsilon = I_n$.

We remark that in this paper, the notion of *Zero Mean Wide Sense Stationary w.r.t. Inputs* (ZMWSSI) process will be a central notion for the mathematical framework of stochastic LPV-SSA representations. We refer the reader to [16, Definition 2] for the detailed definition of ZMWSSI processes. All the process considered in this paper will be assumed to be ZMWSSI process w.r.t. scheduling $\boldsymbol{\mu}$.

Definition 1 (Stationary LPV-SSA): The LPV-SSA representation (1) is *stationary* for $\{\boldsymbol{\mu}_\sigma\}_{\sigma\in\Sigma}$, if

1. The joint state and noise process $\begin{bmatrix}\mathbf{x}^\top & \mathbf{v}^\top\end{bmatrix}^\top$ is ZMWSSI, [16, Definition 2].
2. $\forall w \in \Sigma^+$, $t \in \mathbb{Z}$: $\mathbb{E}\left[\mathbf{v}(t - |w|)\boldsymbol{\mu}_w(t-1)\mathbf{v}^\top(t)\right] = 0$.
3. $\forall w \in \Sigma^+$, $t \in \mathbb{Z}$: $\mathbb{E}\left[\mathbf{x}(t - |w|)\boldsymbol{\mu}_w(t-1)\mathbf{v}^\top(t)\right] = 0$ and for all $\sigma \in \Sigma$, $t \in \mathbb{Z}$: $\mathbb{E}\left[\mathbf{x}(t)\boldsymbol{\mu}_\sigma^2(t)\mathbf{v}^\top(t)\right] = 0$.
4. *Stability:* The eigenvalues of the matrix $\sum_{\sigma\in\Sigma} p_\sigma A_\sigma \otimes A_\sigma$ are inside the open unit circle.

Stationary LPV-SSA representations correspond to stationary generalized bilinear systems in the terminology of [16], if $\boldsymbol{\mu}_\sigma(t)$ is viewed as an input of the bilinear system for all

$\sigma \in \Sigma$. It can be proved (see [16]) that if a stationary LPV-SSA is a realization of $\mathbf{y}$, then $\mathbf{y}$ is ZMWSSI. Note that for stationary LPV-SSA representations, the state process $\mathbf{x}$ is uniquely determined by the system matrices and the noise process $\mathbf{v}$ satisfying Definition 1. This prompts us to use the following short-hand notation for stationary LPV-SSA representations.

Notations 3: A stationary LPV-SSA representation of the form (1) is identified with the tuple $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, D, \mathbf{v})$.

B. Forward innovation form

In the sequel, we will especially be interested in LPV-SSA in the so called forward innovation form.

Notations 4 (Orthogonal projection E_l): We will use the *orthogonal projection* $E_l[\cdot]$ from [16, Notation 3].

We define the *forward innovation process* $\mathbf{e}$ of ZMWSSI $\mathbf{y}$ as $\mathbf{e}(t) = \mathbf{y}(t) - E_l[\mathbf{y}(t) \mid \{\mathbf{z}^{\mathbf{y}}_w(t)\}_{w\in\Sigma^+}]$. Informally, $E_l[\mathbf{y}(t) \mid \{\mathbf{z}^{\mathbf{y}}_w(t)\}_{w\in\Sigma^+}]$ denotes *projection* of $\mathbf{y}$ on its past values $\mathbf{z}^{\mathbf{y}}_w$ and $\mathbf{e}$ is the prediction error of the best linear predictor of $\mathbf{y}$ which is computed based on the products between past outputs and past scheduling signal values.

The LPV-SSA representation of the form (1), is said be a realization of $\mathbf{y}$ in *forward innovation form*, if it is stationary, it is a realization of $\mathbf{y}$ and $D = I_p$ and $\mathbf{v} = \mathbf{e}$, i.e.,

$$\mathbf{x}(t+1)=\sum_{i=1}^{d}\boldsymbol{\mu}_i(t)\left(A_i\mathbf{x}(t)+K_i\mathbf{e}(t)\right), \mathbf{y}(t)=C\mathbf{x}(t)+\mathbf{e}(t). \quad (4)$$

A stationary LPV-SSA is a *minimal* realization of $\mathbf{y}$ if it has the minimal dimension among all possible stationary realizations of $\mathbf{y}$. It turns out that we can restrict attention to LPV-SSA realizations in forward innovation form. To state the result formally, we need to introduce the following definitions. A process $\mathbf{y}$ is said to be *square integrable w.r.t.* $\{\boldsymbol{\mu}_\sigma\}_{\sigma\in\Sigma}$ *(SII process)*, if $\forall w \in \Sigma^+$, the following process $\mathbf{z}^{\mathbf{y}+}$ is square integrable. $\forall t \in \mathbb{Z} : \mathbf{z}^{\mathbf{y}+}(t) = \mathbf{y}(t + |w|)\boldsymbol{\mu}_w(t + |w| - 1)\frac{1}{\sqrt{p_w}}$. Let us call a process $\mathbf{y}$ *full rank*, if its innovation process satisfies the following condition: for all $\sigma \in \Sigma$, the covariance matrix $\mathbb{E}\left[\mathbf{e}(t)\mathbf{e}^T(t)\boldsymbol{\mu}^2_\sigma(t)\right]$ is invertible.

Theorem 1 ([16]): Assume that process $\mathbf{y}$ is SII.

- If $\mathbf{y}$ has a realization by stationary LPV-SSA of the form (1), then it has a minimal LPV-SSA realization $(\{\hat{A}_\sigma, \hat{K}_\sigma\}_{\sigma\in\Sigma}, \hat{C}, I, \mathbf{e})$ in forward innovation form.
- The matrices $\{\hat{A}_\sigma, \hat{K}_\sigma\}_{\sigma\in\Sigma}, \hat{C}$ of this LPV-SSA realization in forward innovation form and the covariances $\{\hat{Q}_i{=}\mathbb{E}\left[\mathbf{e}(t)\mathbf{e}^\top(t)\boldsymbol{\mu}^2_i(t)\right]\}^d_{i=1}$ of the innovation process can be computed from the matrices $\{\{A_i, K_i, Q_i = E[\mathbf{v}(t)\mathbf{v}^T(t)\boldsymbol{\mu}^2_i(t)]\}^d_{i=1}, C, D\}$ of (1).
- If $\mathbf{y}$ is full rank, then the minimal LPV-SSA realizations of $\mathbf{y}$ in forward innovation form are *isomorphic*, i.e. if $(\{\hat{A}_\sigma, \hat{K}_\sigma\}_{\sigma\in\Sigma}, \hat{C}, I, \mathbf{e})$ and $(\{A_\sigma, K_\sigma\}_{\sigma\in\Sigma}, C, I, \mathbf{e})$ are two minimal realizations of $\mathbf{y}$ in forward innovation form, then there exists a non-singular square matrix T, such that $TA_\sigma T^{-1}{=}\hat{A}_\sigma, TK_\sigma{=}\hat{K}_\sigma, \sigma \in \Sigma, CT^{-1}{=}\hat{C}$.

Theorem 1 prompts us to aim at identifying the LPV-SSA representation in a forward innovation form (4).

Problem 2: Assume that $\bar{y} : \mathbb{Z} \to \mathbb{R}^p$ and $\bar{\mu} : \mathbb{Z} \to \mathbb{R}^d$ are sample paths of the processes $\mathbf{y}$ and $\boldsymbol{\mu}$ respectively. Given a dataset $\{\bar{y}(t), \bar{\mu}(t)\}^N_{t=1}$, compute from this dataset the estimates $\{\hat{A}_\sigma, \hat{K}_\sigma, \hat{Q}_\sigma, \hat{C}\}_{\sigma\in\Sigma}$, such that as $N \to \infty$, the estimated matrices $\{\{\hat{A}_\sigma, \hat{K}_\sigma, \hat{Q}_\sigma\}_{\sigma\in\Sigma}, \hat{C}\}$ converge (in a suitable sense) to the system matrices $\{\{A_\sigma, K_\sigma, Q_\sigma\}_{\sigma\in\Sigma}, C\}$, of a minimal realization of output process $\mathbf{y}$ in forward innovation form (4). In particular, for a large enough N, the output $\hat{\mathbf{y}}$ obtained from the estimated matrices $\{\{\hat{A}_\sigma, \hat{K}_\sigma\}_{\sigma\in\Sigma}, \hat{C}\}$, is close to the observed output $\mathbf{y}$ in a suitable sense.

Minimal LPV-SSAs in forward innovation form are unique up to isomorphism, which renders the problem formulation above well-posed. The well-posedness of a similar problem formulation was implicitly assumed in the LPV subspace identification literature.

III. REALIZATION ALGORITHM FOR STOCHASTIC LPV-SSA REPRESENTATION

In this section, we present a computationally efficient realization algorithm for stochastic LPV-SSA representation in forward innovation form (4). This algorithm is a modification of Algorithm 2 presented in [16, page 10], the main difference being the size of the Hankel-matrix used. In order to formulate the announced realization algorithm, we define the *covariance sequence* and *Hankel matrices*.

Let $\mathbf{y} \in \mathbb{R}^p$ be a ZMWSSI stochastic process. Let us define the covariance matrices $\Lambda^{\mathbf{y}}_w \in \mathbb{R}^{p\times p}$, $w \in \Sigma^+$, and $T^{\mathbf{y}}_{\sigma,\sigma} \in \mathbb{R}^{p\times p}$, $\sigma \in \Sigma$, as follows

$$\Lambda^{\mathbf{y}}_w{=}\mathbb{E}\left[\mathbf{y}(t)(\mathbf{z}^{\mathbf{y}}_w(t))^\top\right], \quad T^{\mathbf{y}}_{\sigma,\sigma}{=}\mathbb{E}\left[\mathbf{z}^{\mathbf{y}}_\sigma(t)(\mathbf{z}^{\mathbf{y}}_\sigma(t))^\top\right], \quad (5)$$

where $\mathbf{z}^{\mathbf{y}}_w$ is as defined in (3).

The *covariance sequence* of $\mathbf{y}$ is defined as the function $\Psi_{\mathbf{y}} : \Sigma^* \to \mathbb{R}^{p\times pd}$, such that

$$\Psi_{\mathbf{y}}(w) = [\Lambda^{\mathbf{y}}_{1w}, \Lambda^{\mathbf{y}}_{2w}, \ldots, \Lambda^{\mathbf{y}}_{dw}]. \quad (6)$$

The covariance sequence can be seen as a collection of higher order moments which can determine the matrices of an LPV-SSA representation which realizes $\mathbf{y}$. Conversely, the values of $\Psi_{\mathbf{y}}(w)$ can be expressed in terms of matrices of a LPV-SSA representation (1). To this end, we define the following matrices

$$\begin{aligned} &B_\sigma = \frac{1}{\sqrt{p_\sigma}}(A_\sigma P_\sigma C^\top + K_\sigma Q_\sigma) \quad \forall\sigma \in \Sigma, \\ &B = [B_1, B_2, \ldots, B_d], \\ &P_\sigma = \mathbb{E}\left[\mathbf{x}(t)\mathbf{x}^\top(t)\boldsymbol{\mu}^2_\sigma(t)\right], \; Q_\sigma = \mathbb{E}\left[\mathbf{e}(t)\mathbf{e}^\top(t)\boldsymbol{\mu}^2_\sigma(t)\right] \end{aligned} \quad (7)$$

Note that, by [16] the state-covariance P_σ is the unique solution of the following equation:

$$P_\sigma = p_\sigma \sum_{\sigma_1\in\Sigma}\left(A_{\sigma_1}P_{\sigma_1}A^\top_{\sigma_1} + K_{\sigma_1}Q_{\sigma_1}K^\top_{\sigma_1}\right). \quad (8)$$

Thus, P_σ is uniquely determined by the matrices $\{A_\sigma, K_\sigma, Q_\sigma\}$. Then from [16], the covariances (5), (6) satisfy the following equation: $\Psi_{\mathbf{y}}(w) = \sqrt{p_w}CA_wB$, $\forall w \in \Sigma^*$ and $T^{\mathbf{y}}_{\sigma,\sigma} = \frac{1}{p_\sigma}(CP_\sigma C^\top + Q_\sigma)$, $\forall\sigma \in \Sigma$. Here, we used Notation 2 for A_w.

Next, we introduce the notion of *r-selection*. Let us define the set Σ^r as the set of all words $w \in \Sigma^*$ of length less than or equal to r, i.e., $\Sigma^r = \{w \in \Sigma^* \mid |w| \leq r\}$. We define r-selection as a pair (α, β) such that [15], 1. $\alpha \subseteq \Sigma^r \times \{1, 2, \cdots, p\}$ and $\beta \subseteq \Sigma^r \times \{1, 2, \cdots, p \cdot d\}$; 2. $\mathrm{card}(\alpha) = \mathrm{card}(\beta) = r$, where card denotes cardinality of the set. Thus, α and β are sets such that each element of these sets is a *word-index pair*. We fix the following ordering of α and β.

$$\begin{aligned} \alpha &= \{(u_i, k_i)\}_{i=1}^r,\ u_i \in \Sigma^r, k_i \in \{1, 2, \cdots, p\}, \\ \beta &= \{(v_j, l_j)\}_{j=1}^r,\ v_j \in \Sigma^r, l_j \in \{1, 2, \cdots, p \cdot d\}, \end{aligned} \tag{9}$$

Example 1: Consider $r = 2$, number of outputs $p = 2$, and scheduling signal dimension $d = 2$, we have, $\Sigma^r = \{\epsilon, 1, 2, 11, 12, 21, 22\}$. Then, one of the *r-selection* pair (α, β) can be chosen as, for e.g., $\alpha = \{(u_1, k_1), (u_2, k_2)\} = \{(\epsilon, 1), (11, 2)\}$ and $\beta = \{(v_1, l_1), (v_2, l_2)\} = \{(21, 3), (22, 4)\}$.

Let us now define the Hankel matrix $H_{\Psi,\alpha,\beta} \in \mathbb{R}^{n \times n}$ as follows: $i, j = 1, \ldots, n$, the (i, j)th element of $H_{\Psi,\alpha,\beta}$ is of the form

$$[H_{\Psi,\alpha,\beta}]_{i,j} = [\Psi_{\mathbf{y}}(v_j u_i)]_{k_i, l_j}, \tag{10}$$

where $\Psi_{\mathbf{y}}$ is defined in (6) and $[\Psi_{\mathbf{y}}(v_j u_i)]_{k_i, l_j}$ denotes the entry of $\Psi_{\mathbf{y}}(v_j u_i)$ on the k_ith row and l_jth column, and $(u_i, k_i) \in \alpha, (v_j, l_j) \in \beta$ are as in the ordering of (9). Intuitively, the rows of $H_{\Psi,\alpha,\beta}$ are indexed by word-index pair $(u_i, k_i) \in \alpha$, and similarly, the columns of Hankel matrix $H_{\Psi,\alpha,\beta}$ are indexed by word-index pair $(v_j, l_j) \in \beta$, and the element of $H_{\Psi,\alpha,\beta}$ with the row indexed (u_i, k_i) and column index (v_j, l_j) is the (k_i, l_j)th entry of $\Psi_{\mathbf{y}}(v_j u_i)$.

In addition, we define the σ-shifted Hankel-matrix $Z_{\sigma,\Psi,\alpha,\beta} \in \mathbb{R}^{n \times n}$ as follows: its i, j-th entry is given by

$$[Z_{\sigma,\Psi,\alpha,\beta}]_{i,j} = [\Psi_{\mathbf{y}}(v_j \sigma u_i)]_{k_i, l_j}. \tag{11}$$

Algorithm 1, computes the matrices $(\{\hat{A}_\sigma, \hat{B}_\sigma, \hat{K}_\sigma^M, \hat{Q}_\sigma^M, \hat{P}_\sigma^M\}_{\sigma \in \Sigma}, \hat{C})$, given n-selection (α, β), reduced Hankel matrix $H_{\Psi,\alpha,\beta}$, and the matrices $\{Z_{\sigma,\Psi,\alpha,\beta}\}_{\sigma \in \Sigma}$ and covariances $\{\Psi_{\mathbf{y}}(u_i), \Psi_{\mathbf{y}}(v_i)\}_{i=1}^n$, $\{T_{\sigma,\sigma}^{\mathbf{y}}\}_{\sigma \in \Sigma}$. These matrices can be constructed from *finitely many* ($\mathcal{O}(n^2)$) values of $\Psi_{\mathbf{y}}$. Intuitively, the matrices $\{\hat{A}_\sigma, \hat{B}_\sigma\}_{\sigma \in \Sigma}, \hat{C}$ are such that $\Psi_{\mathbf{y}}(w) = \hat{C} \hat{A}_w \hat{B}$, $\hat{B} = [\hat{B}_1, \hat{B}_2, \ldots, \hat{B}_d]$, and for $\hat{K}_\sigma = \lim_{M \to \infty} \hat{K}_\sigma^M$, $(\{\hat{A}_\sigma, \hat{K}_\sigma\}_{\sigma \in \Sigma}, \hat{C}, I, \mathbf{e})$ is a minimal realization of $\mathbf{y}$ in forward innovation form whose state $\hat{\mathbf{x}}$ satisfies $\hat{P}_\sigma = \mathbb{E}\left[\hat{\mathbf{x}}(t)\hat{\mathbf{x}}^\top(t)\boldsymbol{\mu}_\sigma^2(t)\right] = \lim_{M \to \infty} \hat{P}_\sigma^M$, and $\hat{Q}_\sigma = \mathbb{E}\left[\mathbf{e}(t)\mathbf{e}^\top(t)\boldsymbol{\mu}_\sigma^2(t)\right] = \lim_{M \to \infty} \hat{Q}_\sigma^M$.

This intuition is formalized in the following theorem.

Theorem 2 (Correctness of Algorithm 1): Assume $\mathbf{y}$ is SII, and $\mathbf{y}$ has a realization by stationary stochastic LPV-SSA representation and $\mathbf{y}$ is full rank. Assume that the n-selection pair (α, β) be such that $\mathrm{rank}\ H_{\Psi,\alpha,\beta} = n$, where n is the state-space dimension of a minimal realization. Then Algorithm 1 is well-defined, and the matrices $(\{\hat{A}_\sigma, \hat{B}_\sigma, \hat{K}_\sigma^M, \hat{Q}_\sigma^M, \hat{P}_\sigma^M\}_{\sigma \in \Sigma}, \hat{C})$ returned by Algorithm 1 have the following property: $\hat{K}_\sigma = \lim_{M \to \infty} \hat{K}_\sigma^M$, $\hat{Q}_\sigma = \lim_{M \to \infty} \hat{Q}_\sigma^M$, $\hat{Q}_\sigma^M > 0, \hat{P}_\sigma = \lim_{M \to \infty} \hat{P}_\sigma^M$, and $(\{\hat{A}_\sigma, \hat{K}_\sigma, \}_{\sigma \in \Sigma}, \hat{C}, I, \mathbf{e})$ is a minimal LPV-SSA realization of $\mathbf{y}$ in forward innovation form with $\hat{Q}_\sigma = \mathbb{E}\left[\mathbf{e}(t)\mathbf{e}^\top(t)\boldsymbol{\mu}_\sigma^2(t)\right]$ and the unique state $\hat{\mathbf{x}}$ of this LPV-SSA realization satisfies $\hat{P}_\sigma = \mathbb{E}\left[\hat{\mathbf{x}}(t)\hat{\mathbf{x}}^\top(t)\boldsymbol{\mu}_\sigma^2(t)\right]$, $\sigma \in \Sigma$.

Proof: [Sketch of the proof of Theorem 2] The proof is a combination of the proof of the correctness of the basis reduced Ho-Kalman like realization algorithm in [5] and the proof of correctness of Algorithm 2 in [16, page 10]. ■

Algorithm 1 Matrix computations using $H_{\Psi,\alpha,\beta}$ and n-selection

Input: n-selection (α, β); Hankel matrix $H_{\Psi,\alpha,\beta}$ (defined in (10)); shifted Hankel-matrix $\{Z_{\Psi,\sigma,\alpha,\beta}\}_{\sigma \in \Sigma}$ (defined in (11)); $\{\Psi_{\mathbf{y}}(u_i)\}_{i=1}^n$, $\{\Psi_{\mathbf{y}}(v_i)\}_{i=1}^n$ where $\{u_i, v_i\}_{i=1}^n$ as in (9) with $r = n$, and $T_{\sigma,\sigma}^{\mathbf{y}}$ (defined in (5)), maximum number of iterations $M > 0$.

1. Compute $\hat{A}_\sigma \in \mathbb{R}^{n \times n}$ as linear least-square solution of
$$\hat{A}_\sigma H_{\Psi,\alpha,\beta} = \frac{1}{\sqrt{p_\sigma}} Z_{\sigma,\Psi,\alpha,\beta}.$$
2. Compute $\hat{B}_\sigma \in \mathbb{R}^{n \times p}$ as follows:
Let $\hat{B} \in \mathbb{R}^{n \times p \cdot d}$ be matrix such that its i, j-th element is given by $\left[\hat{B}\right]_{i,j} = [\Psi_{\mathbf{y}}(\epsilon u_i)]_{k_i, j} = [\Psi_{\mathbf{y}}(u_i)]_{k_i, j}$ where $(u_i, k_i) \in \alpha$ is as in (9), $r = n$, and $j = 1, 2, \ldots, p \cdot d$
Then, $\hat{B}_\sigma \in \mathbb{R}^{n \times p}$ for each $\sigma \in \Sigma$ is obtained by appropriate partitioning of B as $\hat{B} = \left[\hat{B}_1, \hat{B}_2, \ldots, \hat{B}_d\right]$.
3. Compute $\hat{C} \in \mathbb{R}^{p \times n}$ as follows:
Let $\bar{C}$ such that its i, j-th element is given by $\left[\bar{C}\right]_{i,j} = [\Psi_{\mathbf{y}}(v_j \epsilon)]_{i, l_j} = [\Psi_{\mathbf{y}}(v_j)]_{i, l_j}$ where $(v_j, l_j) \in \beta$ as in (9), $r = n$, and $i = 1, 2, \ldots, p$. Then, $\hat{C} = \bar{C}(H_{\Psi,\alpha,\beta})^{-1}$.
4. Set $\hat{P}_\sigma^0 = 0$ and **for** $i = 1, \ldots, M$ **do**
$$\begin{aligned} \hat{P}_\sigma^{i+1} &= \sum_{\sigma_1 \in \Sigma} p_\sigma \left(\hat{A}_{\sigma_1} \hat{P}_{\sigma_1}^i \hat{A}_{\sigma_1}^\top + \hat{K}_{\sigma_1}^i \hat{Q}_{\sigma_1}^i (\hat{K}^i)_{\sigma_1}^\top\right) \\ \hat{Q}_\sigma^i &= p_\sigma T_{\sigma,\sigma}^{\mathbf{y}} - \hat{C} \hat{P}_{\sigma_1}^i \hat{C}^\top \\ \hat{K}_\sigma^i &= \left(\hat{B}_\sigma \sqrt{p_\sigma} - \hat{A}_\sigma \hat{P}_\sigma^i \hat{C}^\top\right) \left(\hat{Q}_\sigma^i\right)^{-1} \end{aligned}$$

Output: Matrices $(\{\hat{A}_\sigma, \hat{B}_\sigma, \hat{K}_\sigma^M, \hat{Q}_\sigma^M, \hat{P}_\sigma^M\}_{\sigma \in \Sigma}, \hat{C})$

IV. IDENTIFICATION ALGORITHM FOR STOCHASTIC LPV-SSA REPRESENTATION

In this section, we formulate a computationally efficient identification algorithm based on stochastic realization and selections, using N-length observation sequence of outputs and scheduling signals as detailed in Algorithm 2. In Algorithm 2, the following assumptions are made:

Assumption 1: **(1)** The process $\mathbf{y}$ is SII, full rank, and $\mathbf{y}$ has a realization by stationary stochastic LPV-SSA representation. **(2)** The n-selection pair (α, β) is such that $\mathrm{rank}\ H_{\Psi,\alpha,\beta} = n$, where n is the state-space dimension of a minimal LPV-SSA realization of $\mathbf{y}$. **(3)** The process $(\mathbf{y}, \{\boldsymbol{\mu}_w\}_{w \in \Sigma^+})$ is ergodic and there exist sample paths $\bar{y} : \mathbb{Z} \to \mathbb{R}^p$ and $\bar{\mu} : \mathbb{Z} \to \mathbb{R}^d$ of the processes $\mathbf{y}$ and

$\boldsymbol{\mu}$ respectively such that $\{\bar{y}(t), \{\bar{\mu}_\sigma(t)\}_{\sigma\in\Sigma}\}_{t=1}^N$ is observed and for all $v, w \in \Sigma^+$ the following holds,

$$\mathbb{E}\left[\mathbf{y}(t)(\mathbf{z}_w^{\mathbf{y}}(t))^\top\right] = \lim_{N\to\infty} \frac{1}{N} \sum_{t=|w|}^{N} \bar{y}(t)(\bar{z}_w(t))^\top$$

$$\mathbb{E}\left[\mathbf{z}_v^{\mathbf{y}}(t)(\mathbf{z}_w^{\mathbf{y}}(t))^\top\right] = \lim_{N\to\infty} \frac{1}{N} \sum_{t=\max(|v|,|w|)}^{N} \bar{z}_v(t)\bar{z}_w^\top(t)$$

where for all $w = \sigma_1\sigma_2\cdots\sigma_r \in \Sigma^+$, $r > 0$,

$$\begin{aligned} \bar{\mu}_w(t) &= \bar{\mu}_{\sigma_1}(t-k+1)\bar{\mu}_{\sigma_2}(t-k+2)\cdots\bar{\mu}_{\sigma_r}(t) \\ \bar{z}_w(t) &= \bar{y}(t-|w|)\bar{\mu}_w(t-1)\frac{1}{\sqrt{p_w}} \end{aligned} \tag{12}$$

Algorithm 2 computes the estimates $(\{\hat{A}_\sigma^N, \hat{B}_\sigma^N, \hat{K}_\sigma^{M,N}, \hat{Q}_\sigma^{M,N}, \hat{P}_\sigma^{M,N}\}_{\sigma\in\Sigma}, \hat{C}^N)$ of forward innovation form (4), for a given number of iterations M, from the finite time series data $\{\bar{y}(t), \{\bar{\mu}_\sigma(t)\}_{\sigma\in\Sigma}\}_{t=1}^N$. Algorithm 2 computes the approximations of true system matrices as well as noise covariances such that these approximations converge to the true values as $N, M \to \infty$.

Algorithm 2 Identification of stochastic LPV-SSA models

Input: Observations sequence $\{\bar{y}(t), \{\bar{\mu}_\sigma(t)\}_{\sigma\in\Sigma}\}_{t=1}^N$, and n-selection (α, β); maximum number of iterations $M > 0$.

1. Compute approximate covariances from observations:

$$\Lambda_w^N = \frac{1}{N}\sum_{t=|w|}^{N} \bar{y}(t)\bar{z}_w(t); T_{\sigma,\sigma}^N = \frac{1}{N}\sum_{t=|w|}^{N} \bar{z}_\sigma(t)\bar{z}_\sigma^\top(t)$$

   for all $\sigma \in \Sigma$ and for every $w \in \Sigma^+$, such that $w = ivu$ or $w = iv$ or $w = iu$ or $w = iv\sigma u$ for some words $v, u \in \Sigma^*$, $i, \sigma \in \Sigma$, $(u, k) \in \alpha$, $(v, l) \in \beta$ for some $k = 1, \ldots, p$, $l = 1, \ldots, pd$, and for all $w \in \Sigma^+$, $\bar{z}_w(t)$ is as in (12).
2. Construct the *empirical* covariance sequences $\Psi_{\mathbf{y}}^N(w)$ by replacing Λ_w by Λ_w^N in (6), and construct the *empirical* Hankel matrix $H_{\Psi,\alpha,\beta}^N \in \mathbb{R}^{n\times n}$ and the empirical σ shifted Hankel matrix $Z_{\sigma,\Psi,\alpha,\beta}^N \in \mathbb{R}^{n\times n}$, $\sigma \in \Sigma$, by replacing $\Psi_{\mathbf{y}}$ by $\Psi_{\mathbf{y}}^N$ in (10) and (11) respectively, i.e., for all $i, j = 1, \ldots, n$,

$$\begin{aligned} \Psi_{\mathbf{y}}^N(w) &= \left[\Lambda_{1w}^N, \Lambda_{2w}^N, \ldots, \Lambda_{dw}^N\right] \\ \left[H_{\Psi,\alpha,\beta}^N\right]_{i,j} &= \left[\Psi_{\mathbf{y}}^N(v_j u_i)\right]_{k_i,l_j} \\ \left[Z_{\sigma,\Psi,\alpha,\beta}^N\right]_{i,j} &= \left[\Psi_{\mathbf{y}}^N(v_j\sigma u_i)\right]_{k_i,l_j} \end{aligned}$$

3. Run Algorithm 1, with: (1) the empirical Hankel matrix $H_{\Psi,\alpha,\beta}^N$ instead of $H_{\Psi,\alpha,\beta}$; (2) the empirical shifted Hankel matrix $Z_{\sigma,\Psi,\alpha,\beta}^N$ instead of $Z_{\sigma,\Psi,\alpha,\beta}$; (3) the empirical $\{\Psi_{\mathbf{y}}^N(u_i)\}_{i=1}^n$, $\{\Psi_{\mathbf{y}}^N(v_i)\}_{i=1}^n$ instead of $\{\Psi_{\mathbf{y}}(u_i)\}_{i=1}^n$, $\{\Psi_{\mathbf{y}}(v_i)\}_{i=1}^n$ where $\{u_i, v_i\}_{i=1}^n$ as in (9) with $r = n$; (4). the empirical covariance $T_{\sigma,\sigma}^N$ instead of $T_{\sigma,\sigma}^{\mathbf{y}}$, $\forall\sigma \in \Sigma$.

Output: Estimates $\{\hat{A}_\sigma^N, \hat{B}_\sigma^N, \hat{K}_\sigma^{M,N}, \hat{Q}_\sigma^{M,N}, \hat{P}_\sigma^{M,N}\}_{\sigma\in\Sigma}, \hat{C}^N$

Theorem 3 (Consistency): Under Assumption 1, the result of Algorithm 2 satisfies the following for all $\sigma \in \Sigma$:

$$\hat{K}_\sigma = \lim_{M\to\infty}\lim_{N\to\infty} \hat{K}_\sigma^{M,N},\ \hat{Q}_\sigma = \lim_{M\to\infty}\lim_{N\to\infty} \hat{Q}_\sigma^{M,N},$$
$$\hat{P}_\sigma = \lim_{M\to\infty}\lim_{N\to\infty} \hat{P}_\sigma^{M,N},\ \hat{A}_\sigma = \lim_{N\to\infty} \hat{A}_\sigma^N,\ \ \hat{C} = \lim_{N\to\infty} \hat{C}^N,$$

and the LPV-SSA $(\{\hat{A}_\sigma, \hat{K}_\sigma\}_\sigma, \hat{C}, \hat{\mathbf{x}}, \mathbf{e})$ is a minimal realization of $\mathbf{y}$ in forward innovation form.

Proof: [Sketch of the proof of Theorem 3] By the Part **(3)** of Assumption 1, $\lim_{N\to\infty} H_{\Psi,\alpha,\beta}^N = H_{\Psi,\alpha,\beta}$ and $\lim_{N\to\infty} Z_{\sigma,\Psi,\alpha,\beta}^N = Z_{\sigma,\Psi,\alpha,\beta}$. Since $\hat{A}_\sigma^N, \hat{C}^N, \hat{B}_\sigma^N$ are continuous functions of the entries of $H_{\Psi,\alpha,\beta}^N$, $Z_{\sigma,\Psi,\alpha,\beta}^N$, $\sigma \in \Sigma$, it follows that $\hat{A}_\sigma = \lim_{N\to\infty} \hat{A}_\sigma^N$, $\hat{B}_\sigma = \lim_{N\to\infty} \hat{B}_\sigma^N$, $\hat{C} = \lim_{N\to\infty} \hat{C}^N$, where $\{\hat{A}_\sigma, \hat{B}_\sigma\}_{\sigma\in\Sigma}, \hat{C}$ are the matrices computed in Steps 1,2,3 of Algorithm 1 using $H_{\Psi,\alpha,\beta}$, $Z_{\sigma,\Psi,\alpha,\beta}$, $\{\Psi_{\mathbf{y}}(u_i)\}_{i=1}^n$, $\{\Psi_{\mathbf{y}}(v_i)\}_{i=1}^n$ where $\{u_i, v_i\}_{i=1}^n$ as in (9) with $r = n$. It can be shown by induction on M that for all M, $\hat{K}_\sigma^{M,N}$, $\hat{Q}_\sigma^{M,N}$, $\hat{P}_\sigma^{M,N}$ are continuous functions of $\hat{A}_\sigma^N$, $\hat{B}_\sigma^N$, $\hat{C}^N$, and in fact $\hat{K}_\sigma^M = \lim_{N\to\infty} \hat{K}_\sigma^{M,N}$, $\hat{Q}_\sigma^M = \lim_{N\to\infty} \hat{Q}_\sigma^{M,N}$, $\hat{P}_\sigma^M = \lim_{N\to\infty} \hat{P}_\sigma^{M,N}$, $\sigma \in \Sigma$, where $\{\hat{K}_\sigma^M, \hat{Q}_\sigma^M, \hat{P}_\sigma^M\}_{\sigma\in\Sigma}$ are the matrices calculated in Step 4 of Algorithm 1. The statement of Theorem 3 is then a direct consequence of Theorem 2. ■

V. NUMERICAL EXAMPLE

In this section, we present a numerical example to test the effectiveness of our algorithm. The proposed algorithm is compared to the *Predictor Based Subspace IDentification* (PBSID) algorithm reported in [21]. We remark that the PBSID algorithm, requires the observability assumption of the first local model. The realization algorithm proposed in this contribution relaxes this assumption. However, it relies on choosing the correct *selections* (α, β). Specifically, in the following example, we show that the observability assumption is not satisfied by the true data generating system, yet, the proposed algorithm is able to estimate the model matrices such that the estimated output matches closely with the observed outputs. The quality of the match between estimated and true outputs is quantified on a noise-free validation data of length N_{val} via *Best Fit Rate* (BFR) and *Variance Accounted For* (VAF) criterion defined for each output channel y_i, $i = 1, \ldots, p$, as $\text{BFR}_{y_i} = \max\left\{1 - \sqrt{\frac{\sum_{t=1}^{N_{\text{val}}}(y_i(t)-\hat{y}_i(t))^2}{\sum_{t=1}^{N_{\text{val}}}(y_i(t)-\bar{y}_i)^2}}, 0\right\} \times 100\%$, $\text{VAF}_{y_i} = \max\left\{1 - \frac{\text{var}(y_i-\hat{y}_i)}{\text{var}(y_i)}, 0\right\} \times 100\%$, where $\hat{y}_i$ denotes the simulated one-step ahead model output and $\bar{y}_i$ denotes the sample mean of the output over the validation set. The operator $\text{var}(\cdot)$ denotes the variance of its argument.

The LPV-SSA representation in forward innovation form (4) is used for data generation with following matrices:

$$A_1 = \begin{bmatrix} 0.4 & 0.4 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, A_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.4 & 0.4 \\ 0 & 0.4 & 0.4 \end{bmatrix},$$

$$K_1 = \begin{bmatrix} -0.036 \\ 0 \\ 1 \end{bmatrix}, K_2 = \begin{bmatrix} 0 \\ 0.015 \\ 1.17 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix},$$

TABLE I

BEST FIT RATE (BFR) AND VARIANCE ACCOUNTED FOR (VAF) ON A NOISE-FREE VALIDATION DATA

	Algorithm 2	PBSID [21]	PBSID (external inputs)
BFR	93.98 %	1.58 %	0 %
VAF	99.74 %	3.16 %	0 %

which corresponds to state-dimension $n=3$, output dimension $p = 1$, and scheduling signal dimension $d = 2$ with $\Sigma = \{1, 2\}$. Note that, the system corresponding to the first local model $\tilde{A}_1 = A_1 - K_1 C$ is *not observable*, i.e., $\text{rank}([C^\top (C\tilde{A}_1)^\top \dots (C\tilde{A}_1^{l-1})^\top]^\top) = 2 < n$.

Training and noise free validation output sequences of length $N = 500,000$ and $N_{\text{val}} = 20000$, respectively, are generated from (4) by considering the scheduling signal process $\boldsymbol{\mu} = [\boldsymbol{\mu}_1 \;\; \boldsymbol{\mu}_2]$ such that $\boldsymbol{\mu}_1(t) = 1$ and $\boldsymbol{\mu}_2(t)$ is a white-noise process with uniform distribution $\mathcal{U}(-1.5, 1.5)$. This corresponds to the parameter values $\{p_\sigma\}_{\sigma\in\{1,2\}}$ to be $p_1 = \mathbb{E}\left[\boldsymbol{\mu}_1^2(t)\right] = 1$ and $p_2 = \mathbb{E}\left[\boldsymbol{\mu}_2^2(t)\right] = 0.75$. The standard deviation of the white Gaussian noise $\mathbf{e}$ corrupting the training output is 1, i.e., $\mathbf{e} \sim \mathcal{N}(0, 1)$.

We run the Algorithm 2 with $M = 50$ iterations and with the following r-selection pair (α, β), with $r = 3$, $\alpha = \{(\epsilon, 1), (1, 1), (21, 1)\}$, $\beta = \{(\epsilon, 1), (2, 1), (21, 1)\}$, which is used to choose corresponding entries of the Hankel matrix. The mean time taken to run the algorithm is 1.5 sec.[1]

The validation result using one-step ahead predicted outputs $\hat{y}$ are reported in Table I. We compare our algorithm with the PBSID approach reported in [21] and implemented using PBSID MATLAB toolbox. For PBSID, we consider past and future window length 8 with zero external input signal as well as with a white Gaussian noise external input $u \sim \mathcal{N}(0, 1)$ with input matrices $B_1 = B_2 = [0\ 0\ 1]^\top$. The proposed approach significantly outperforms the PBSID method. As the data generating LPV-SSA representation does not satisfy the underlying assumptions of PBSID method, namely non-observability of the first local model A_1, the PBSID method fails.

VI. CONCLUSIONS AND FUTURE WORKS

In this paper, an identification algorithm for stochastic linear parameter-varying state-space representations has been presented. The proposed algorithm provides a computationally efficient alternative to the parametric subspace approaches which are prone to computational and dimensionality problems as the size of the dataset or the dimension of state-space grows. Moreover, the proposed algorithm is provenly consistent. As a future research direction, we would like to extend the results to a more general LPV model class with external inputs processes.

REFERENCES

[1] A. A. Bachnas, R. Tóth, J. Ludlage, and A. Mesbah. A review on data-driven linear parameter-varying modeling approaches: A high-purity distillation column case study. *Journal of Process Control*, 24(4):272–285, 2014.

[2] B. A. Bamieh and L. Giarré. Identification of linear parameter-varying models. *International Journal of Robust Nonlinear Control*, 12(9):841–853, 2002.

[3] M. Batu, M. Petreczky, R. Tóth, R. Wisniewski, J. Leth, and D. Efimov. Moment matching based model reduction for LPV state-space models. In *Proc. 54th IEEE Conference on Decision and Control*, pages 5334–5339, Osaka, Japan, 2015.

[4] V. Cerone, D. Piga, and D. Regruto. Set-membership LPV model identification of vehicle lateral dynamics. *Automatica*, 47(8):1794–1799, 2011.

[5] P. Cox, M. Petreczky, and R. Tóth. Towards efficient maximum likelihood estimation of LPV-SS models. *Automatica*, 97(9):392–403, 2018.

[6] P. Cox, R. Tóth, and M. Petreczky. Estimation of LPV-SS models with static dependency using correlation analysis. In *Proc. 1st IFAC Workshop on Linear Parameter Varying Systems*, pages 91–96, Grenoble, France, 2015.

[7] P.L. dos Santos, J.A. Ramos, and J.L.M. de Carvalho. Identification of bilinear systems with white noise inputs: An iterative deterministic-stochastic subspace approach. *IEEE Transactions on Control Systems Technology*, 17(5):1145–1153, Sept 2009.

[8] W. Favoreel, B. De Moor, and P. Van Overschee. Subspace identification of bilinear systems subject to white inputs. *IEEE Transactions on Automatic Control*, 44(6):1157–1165, Jun 1999.

[9] F. Felici, J. W. Van Wingerden, and M. Verhaegen. Subspace identification of MIMO LPV systems using a periodic scheduling sequence. *Automatica*, 43:1684–1697, 2007.

[10] A. Lindquist and G. Picci. *Linear Stochastic Systems: A Geometric Approach to Modeling, Estimation and Identification*. Springe Berlin, 2015.

[11] A. Marcos and G.J. Balas. Development of linear-parameter-varying models for aircraft. *Journal of Guidance Control and Dynamics*, 27(2):218–228, 2004.

[12] M. Mejari, D. Piga, and A. Bemporad. A bias-correction method for closed-loop identification of Linear Parameter-Varying systems. *Automatica*, 87:128–141, 2018.

[13] C. Novara, F. Ruiz, and M. Milanese. Direct identification of optimal SM-LPV filters and application to vehicle yaw rate estimation. *IEEE Transactions on Control Systems Technology*, 19(1):5–17, 2011.

[14] M. Petreczky, R. Tóth, and G. Mercére. Realization theory for LPV state-space representations with affine dependence. *IEEE Transactions on Automatic Control*, 62(9):4667–4674, 2018.

[15] M. Petreczky and R. Vidal. Realization theory of stochastic jump-markov linear systems. *Arxiv reoprt, arXiv:1412.5020*, 2014.

[16] M. Petreczky and R. Vidal. Realization theory for a class of stochastic bilinear systems. *IEEE Transactions on Automatic Control*, 63(1):69–84, 2018.

[17] D. Piga, P. Cox, R. Tóth, and V. Laurain. LPV system identification under noise corrupted scheduling and output signal observations. *Automatica*, 53:329–338, 2015.

[18] C. W. Scherer. Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control for time-varying and linear parametrically-varying systems. *Int. Journal of Robust and Nonlinear Control*, 6(9-10):929–952, 1996.

[19] M. Tanelli, D. Ardagna, and M. Lovera. Identification of LPV state space models for autonomic web service systems. *IEEE Transactions on Control Systems Technology*, 19(1):93–103, 2011.

[20] R. Tóth. *Modeling and identification of linear parameter-varying systems*. Lecture Notes in Control and Information Sciences, Vol. 403, Springer, Heidelberg, 2010.

[21] J. W. van Wingerden and M. Verhaegen. Subspace identification of bilinear and LPV systems for open-and closed-loop data. *Automatica*, 45(2):372–381, 2009.

[22] V. Verdult and M. Verhaegen. Kernel methods for subspace identification of multivariable LPV and bilinear systems. *Automatica*, 41:1557–1565, 2005.

[23] K. Zhou and J. C. Doyle. *Essentials of Robust Control*. Prentice-Hall, 1998.

[1] All computations are carried out on an i5 1.8-GHz Intel core processor with 8 GB of RAM running MATLAB R2018a.