LPV State-Space Model Identification in the Bayesian Setting: A 3-step Procedure*

Pepijn B. Cox[1] and Roland Tóth[1]

Abstract— Current state-of-the-art linear parameter-varying (LPV) control design methods presume that an LPV state-space (SS) model of the system with affine dependence on the scheduling variable is available. However, many existing LPV-SS identification schemes either suffer heavily from computational issues related to the curse of dimensionality or are based on severe approximations. To overcome these issues, in this paper, the Bayesian framework is combined with a recently developed efficient SS realization scheme. We propose a computationally attractive 3-step approach for identifying LPV-SS models. In Step 1, the sub-Markov parameters representing the impulse response of the system are estimated in a Bayesian setting, using kernel based Ridge regression with hyper-parameter tuning via marginal likelihood optimization. Subsequently, in Step 2, an LPV-SS realization is obtained by using an efficient basis reduced Ho-Kalman like deterministic SS realization scheme on the identified impulse response. Finally, in Step 3, to reach the maximum likelihood estimate, the LPV-SS model is refined by applying a Bayesian expectation-maximization method. The performance of the proposed 3-step scheme is demonstrated on a Monte-Carlo simulation study.

I. INTRODUCTION

The *linear parameter-varying* (LPV) modeling and control framework has received considerable attention in recent years, e.g., see [1]–[3]. The LPV model class offers a flexible framework to capture non-stationary or nonlinear behavior of physical or chemical processes by a varying linear structure [4], [5]. The majority of the powerful LPV control synthesis approaches assume that a *state-space* (SS) model of the system is available, in particular, with static and affine dependence of the involved matrix coefficients on the scheduling variable p, e.g., see [3]. Hence, a central problem in LPV system identification is to estimate such models efficiently based on experimental data. Conceptually, the identification can be accomplished as: i) a direct model estimation problem in the so-called *global identification setting*, which requires experimental data with a varying, informative p; or ii) as the interpolation of local LTI models estimated from multiple experiments around fixed operating points, i.e., constant p, also known as the *local identification setting*. In this paper, we restrict our focus to the global setting, see [5] for a comparison between the two settings.

Due to the success of realization theory based LTI-SS model estimation approaches, also refereed to as *subspace identification methods* (SIMs), various extensions have been developed in the global LPV identification setting: see [6]–[8] to mention a few. Unfortunately, these approaches suffer heavily from the curse of dimensionality and can result in ill-conditioned estimation problems with overwhelming computational and memory needs for real-world applications. Hence, various approximations have been proposed for these methods, but at a heavy price of achievable accuracy [9]. Other approaches, like *set membership* methods, e.g., [10], share the same curse of dimensionality, while nonlinear optimization approaches based on the *prediction error minimization* (PEM) framework result in a heavy nonlinear optimization problem, prone to local minima, e.g., [11]. Recently, a different perspective to the aforementioned gradient-based schemes has appeared to obtain a *maximum likelihood* (ML) estimate in a Bayesian setting. The *expectation-maximization* (EM) method, commonly used in signal processing, has been extended in [12] to identify LPV-SS models by considering the state sequence as missing data. Despite of its ML efficiency, the EM method still requires an efficient initialization due to its iterative steps.

As we show in this paper, efficient initialization of the EM approach can be achieved by applying the core concept of the Bayesian setting, but using an alternative representation of the system dynamics to avoid direct identification of the SS matrix functions. Recent developments of PEM based estimation of *infinite impulse response* (IIR) models in a Bayesian framework offers the possibility to efficiently handle variance issues and allow for maximum likelihood efficiency with low computational requirements [13]–[15]. However, realizing LPV-SS models from LPV-IIR models is, in general, computationally expensive and will result in rational and dynamic dependence of the matrix functions on the scheduling variable or in a non-minimal state realization [2]. Yet, recent developments in LPV realization theory have significantly decreased the computational complexity using a basis reduced Ho-Kalman like realization scheme [16].

In this paper, after discussing the problem setting and the notation (Sec. II), we combine the aforestated methods, to develop a novel 3-step method for identifying LPV-SS models from data. Step 1 corresponds to the estimation of the Markov coefficient sequence of the system in the Bayesian setting (Sec. III). To allow numerically reliable tuning of the hyper parameters of this method, the robust marginal likelihood optimization detailed in [13] is extended to the *multiple-input multiple-output* (MIMO) case (Appx.). In Step 2, the LPV-SS model is realized from the estimated coefficients by using a basis reduced Ho-Kalman like method [16] (Sec. IV). At last, in Step 3, the LPV-SS realization is refined

[1]P.B. Cox and R. Tóth are with the Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, Postbus 513, 5600 MB Eindhoven, The Netherlands. Email: `p.b.cox@tue.nl,r.toth@tue.nl`.

(used as an initialization) by the EM approach to find an ML estimate (Sec. V). The performance of the 3-step method is assessed by a Monte-Carlo study (Sec. VI), followed by the conclusions (Sec. VII).

II. PROBLEM SETTING

A. The data-generating system

Consider a discrete-time, linear, parameter-varying data-generating system given by the *LPV-SS representation*:

$$qx = \mathscr{A}(p)x + \mathscr{B}(p)u + v_x, \tag{1a}$$

$$y = \mathscr{C}(p)x + \mathscr{D}(p)u + v_y, \tag{1b}$$

where $x : \mathbb{Z} \to \mathbb{X} = \mathbb{R}^{n_x}$ is the state variable, $y : \mathbb{Z} \to \mathbb{Y} = \mathbb{R}^{n_y}$ is the measured output signal, $u : \mathbb{Z} \to \mathbb{U} = \mathbb{R}^{n_u}$ denotes the input signal, $p : \mathbb{Z} \to \mathbb{P} \subseteq \mathbb{R}^{n_p}$ is the scheduling variable, q is the forward time-shift operator, e.g., $qx(t) = x(t+1)$ where $t \in \mathbb{Z}$ is the discrete time, and $v_x : \mathbb{Z} \to \mathbb{R}^{n_x}$, $v_y : \mathbb{Z} \to \mathbb{R}^{n_y}$ are zero-mean i.i.d. stationary noise processes satisfying

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} \sim \mathcal{N}(0, \Sigma_v), \qquad \Sigma_v = \begin{bmatrix} \mathcal{Q} & \mathcal{S} \\ \mathcal{S}^\top & \mathcal{R} \end{bmatrix}, \tag{2}$$

where $\mathcal{Q} \in \mathbb{R}^{n_x \times n_x}$, $\mathcal{S} \in \mathbb{R}^{n_x \times n_y}$, and $\mathcal{R} \in \mathbb{R}^{n_y \times n_y}$ are covariance matrices. All signals are assumed to have left compact support and u, p are considered to be independent of v_x and v_y. As common in the control literature [3], the matrix functions $\mathscr{A}(\cdot), ..., \mathscr{D}(\cdot)$ defining the SS representation (1) are assumed to be affine combinations of linearly independent, bounded scalar functions $\psi^{[i]} : \mathbb{P} \to \mathbb{R}$ in p:

$$\mathscr{A}(p) = A_0 + \sum_{i=1}^{n_\psi} A_i \psi^{[i]}(p), \quad \mathscr{B}(p) = B_0 + \sum_{i=1}^{n_\psi} B_i \psi^{[i]}(p),$$

$$\mathscr{C}(p) = C_0 + \sum_{i=1}^{n_\psi} C_i \psi^{[i]}(p), \quad \mathscr{D}(p) = D_0 + \sum_{i=1}^{n_\psi} D_i \psi^{[i]}(p), \tag{3}$$

where $\{A_i, B_i, C_i, D_i\}_{i=0}^{n_\psi}$ are constant matrices with appropriate dimensions. The system (1) with functional dependencies (3) is assumed to be structurally state observable and state reachable[1], which implies joint minimality of (1).

B. Problem statement

We are interested in identifying the constant matrices of the affine dependency structure (3) given by

$$\Omega_0 = \begin{bmatrix} A_0 & \dots & A_{n_\psi} & B_0 & \dots & B_{n_\psi} \\ C_0 & \dots & C_{n_\psi} & D_0 & \dots & D_{n_\psi} \end{bmatrix}, \tag{4}$$

with $\Omega_0 \in \mathbb{R}^{n_x + n_y \times (n_x + n_u)(1 + n_\psi)}$. As is generally known, state-space models cannot be uniquely identified nor realized based upon their input-scheduling-output mapping. A model $\mathscr{M}(\Omega)$ with parametrization according to (4) is isomorphic to the true model $\mathscr{M}(\Omega_0)$. Under the affine dependency structure (3), Ω lies within the following set

$$\mathscr{H} = \left\{ \begin{bmatrix} T^{-1} & 0 \\ 0 & I_{n_y} \end{bmatrix} \Omega_0 \begin{bmatrix} I_{1+n_\psi} \otimes T & 0 \\ 0 & I_{n_u(1+n_\psi)} \end{bmatrix} \right\}, \tag{5}$$

where $T \in \mathbb{R}^{n_x \times n_x}$ is a non-singular matrix and $\otimes$ is the Kronecker product. Hence, a realization $\Omega \in \mathscr{H}$ with LPV-SS model $\mathscr{M}(\Omega)$ has equivalent input-scheduling-output behaviour as $\mathscr{M}(\Omega_0)$ [2].

---

[1]There exists at least one $p \in \mathbb{P}^{\mathbb{Z}}$ such that n_x-step observability and reachability holds for all time moments on the support of the signals [2].

Our objective is to efficiently estimate a realization Ω and, accordingly, the minimal state dimension (dynamic order) n_x of the underlying system, based on a data set $\mathcal{D}_N = \{u(t), p(t), y(t)\}_{t=1}^N$ generated by (1).

C. Impulse response representation

To be able to solve our problem as given in the previous section, we make use of an alternative representation of (1):

Lemma 1 (IIR representation [2]): Any asymptotically stable[2] LPV system given in terms of representation (1) has a convergent series expansion in terms of the pulse-basis $\{q^{-i}\}_{i=0}^\infty$ given by

$$y = \sum_{i=0}^\infty (h_i \diamond p) q^{-i} u + w, \tag{6}$$

where $h_i \in \mathscr{R}^{n_y \times n_u}$ are the expansion coefficient functions, i.e., *Markov coefficients*, $\mathscr{R}$ defines the ring of all real polynomial functions with finite dimensional domain and essential support, the operator $\diamond : (\mathscr{R}, \mathbb{P}^{\mathbb{Z}}) \to (\mathbb{R}^{n_y \times n_u})^{\mathbb{Z}}$ denotes $(h_i \diamond p) = h_i(p(t), \dots, p(t - \tau))$ with $\tau \in \mathbb{Z}$ according to $\dim(\mathrm{dom}(h_i))$, w is a zero mean, quasi-stationary noise process. □

In (6), the IIR coefficients $\{(h_i \diamond p)\}_i^\infty$ are given by

$$y = \underbrace{\mathscr{D}(p)}_{h_0 \diamond p} u + \underbrace{\mathscr{C}(p)\mathscr{B}(q^{-1}p)}_{h_1 \diamond p} q^{-1} u +$$

$$\underbrace{\mathscr{C}(p)\mathscr{A}(q^{-1}p)\mathscr{B}(q^{-2}p)}_{h_2 \diamond p} q^{-2} u + \cdots + w, \tag{7}$$

where h_i converges to the zero function as $i \to \infty$. The noise w in (6) is colored, dependent on p and contains the filtered process noise v_x and the additive output noise v_y of (1). For notional ease, we introduce $\psi^{[i]}(p) = \psi^{[i]}$ and $\psi = \begin{bmatrix} 1 & \psi^{[1]} & \cdots & \psi^{[n_\psi]} \end{bmatrix}^\top$. Applying this notation, the Markov coefficients can be written as

$$h_m \diamond p = \mathscr{C}(p)\mathscr{A}(q^{-1}p) \cdots \mathscr{A}(q^{-(m-1)}p)\mathscr{B}(q^{-m}p) =$$

$$\sum_{i=0}^{n_\psi} \sum_{j=0}^{n_\psi} \cdots \sum_{k=0}^{n_\psi} \sum_{l=0}^{n_\psi} C_i A_j \cdots A_k B_l \psi_i \overleftarrow{\psi}_1^{[j]} \cdots \overleftarrow{\psi}_m^{[l]}, \tag{8}$$

where the individual products $C_i A_j \cdots A_k B_l$ are called the *sub-Markov parameters* and $\overleftarrow{[\cdot]}_\tau$ denotes a signal shifted τ steps backwards in time, i.e., $\overleftarrow{\psi}_\tau^{[i]}$ represents $\psi^{[i]}(q^{-\tau}p)$. Note that the Markov coefficients h_i in (7) are independent of the parametrization of the matrix functions, while the sub-Markov parameters are dependent on the parameterization of the functional dependencies in (3). Due to the convergence of h_i, it is often sufficient to truncate (6) to a certain order:

$$y \approx \sum_{i=0}^{n_h} (h_i \diamond p) q^{-i} u + w, \tag{9}$$

where $n_h > 0$ is the order of the resulting *finite impulse response* (FIR) model.

---

[2]An LPV system, represented in terms of (1) with $v_x = 0$ and $v_y = 0$, is called asymptotically stable if, for all trajectories of $(u(t), p(t), y(t))$ satisfying (1), with $u(t) = 0$ for $t \geq 0$ and $p(t) \in \mathbb{P}$, it holds that $\lim_{t \to \infty} |y(t)| = 0$.

III. Bayesian Impulse Response Estimation

The first step in the proposed identification scheme is to identify the sub-Markov parameters based upon data. In this paper, a Ridge regression bases LPV-FIR estimation procedure is used, where the optimal regularization matrix is determined in a Bayesian way with Gaussian prior. How the Bayesian framework and regularized ℓ_2 framework connect, e.g., see [17]. In addition, the Bayesian framework also allows to estimate the functional dependencies ψ in a nonparamtric way [15]. However, for the sake of simplicity, it is assumed that these functions are known a priori.

A. The truncated IIR model

In this section, it is assumed that the process noise is zero, i.e., $v_x = 0$ in (1). Hence, the output additive noise v_y in (1) is equal to the output additive noise w in (6) and (9) corresponding to an output error setting. We will return to the effect of this assumption later. For notational ease, the time of the signals will be indicated by a subscript, e.g., u_3 refers to $u(3)$. Furthermore, define

$$\mathcal{M}_1 = \begin{bmatrix} B_0 & \cdots & B_{n_\psi} \end{bmatrix}, \quad \mathcal{M}_j = \begin{bmatrix} A_0\mathcal{M}_{j-1} & \cdots & A_{n_\psi}\mathcal{M}_{j-1} \end{bmatrix}. \quad (10)$$

Based on (8) and (9), the samples in $\mathcal{D}_N$ satisfy the following relationship:

$$\begin{bmatrix} y_{n_h+1} & \ldots & y_N \end{bmatrix} = \begin{bmatrix} w_{n_h+1} & \ldots & w_M \end{bmatrix} +$$
$$\begin{bmatrix} D_0 & \ldots & D_{n_\psi} & C_0\mathcal{M}_1 & \ldots & C_{n_\psi}\mathcal{M}_1 & C_0\mathcal{M}_2 & \ldots & C_{n_\psi}\mathcal{M}_{n_h} \end{bmatrix}$$
$$\begin{bmatrix} \psi_{n_h+1} \otimes u_{n_h+1} & \cdots & \psi_N \otimes u_N \\ \psi_{n_h+1} \otimes \psi_{n_h} \otimes u_{n_h} & \cdots & \psi_N \otimes \psi_{N-1} \otimes u_{N-1} \\ \vdots & \ddots & \vdots \\ \psi_{n_h+1} \otimes \ldots \otimes \psi_1 \otimes u_1 & \ldots & \psi_N \otimes \ldots \otimes \psi_{N-n_h} \otimes u_{N-n_h} \end{bmatrix},$$

or, written more compactly as

$$\overline{Y}_N = \overline{\theta}_0 \, \overline{\Phi}_N + \overline{W}_N, \quad (11)$$

where $M = N - n_h - 1$, $\overline{Y}_N \in \mathbb{R}^{n_y \times M}$ are the measured outputs, $\overline{\theta}_0 \in \mathbb{R}^{n_y \times \sum_{i=1}^{n_h+1}(1+n_\psi)^i n_u}$ is the matrix with sub-Markov parameters, $\overline{\Phi}_N \in \mathbb{R}^{\sum_{i=1}^{n_h+1}(1+n_\psi)^i n_u \times M}$ is the regression matrix and $\overline{W}_N \in \mathbb{R}^{n_y \times M}$ the cumulative noise process. The resulting output predictor of the MIMO FIR model (11) is

$$\hat{Y}_N = \Phi_N^\top \theta, \quad (12)$$

where $n_\theta = n_y \sum_{i=1}^{n_h+1}(1+n_\psi)^i n_u$, $\hat{Y}_N \in \mathbb{R}^{n_y M \times 1}$ is the predicted output, $\Phi_N^\top = \overline{\Phi}_N^\top \otimes I_{n_y} \in \mathbb{R}^{n_y M \times n_\theta}$, and $\theta \in \mathbb{R}^{n_\theta \times 1}$. For notational reasons also introduce $Y_N = \text{vec}(\overline{Y}_N)$, $\theta_0 = \text{vec}(\overline{\theta}_0)$, and $W_N = \text{vec}(\overline{W}_N)$.

B. Ridge regression based estimate

A well-know issue in estimation of FIR models via the least-squares approach is the high variance of the estimated parameters, even in the SISO LTI case, due to the relatively large number of parameters required to adequately represent (1). ℓ_2 Regularization makes it possible to control the so called *bias-variance trade-off*, i.e., dramatically decrease the variance by introducing a relatively small bias on the estimates [17]. The *weighted Ridge regression* problem is

$$\min_\theta \|\Phi_N^\top \theta - Y_N\|_{W_e}^2 + \|\theta\|_{W_r}^2, \quad (13)$$

where $\|x\|_W = \sqrt{x^\top W x}$ denotes the *weighted Euclidean norm*, $W_e, W_r \in \mathbb{R}^{n_\theta \times n_\theta}$ are positive semi-definite (symmetric) regularization matrices and the analytic solution of (13) is

$$\hat{\theta}_{\text{RWLS}} = \left(\Phi_N W_e \Phi_N^\top + W_r \right)^{-1} \Phi_N W_e Y_N. \quad (14)$$

The regularization matrix W_r is chosen such that $\Phi_N W_e \Phi_N^\top + W_r$ is invertible. If $W_r = 0$, then (14) is the least squares solution, which results in the asymptotically efficient, unbiased, ML estimate. However, if $W_r = 0$ and the additive noise w is a zero mean colored noise, but uncorrelated with the input signal then the estimator is still unbiased, although, it will result in an inefficient estimator with increased variance. If w and u are correlated then an LPV *instrumental variable* (IV) estimator can be used to remove the bias, e.g., see [18]. According to our previous assumptions, in this section, we restrict ourselves to an i.i.d. $w = v_y \sim \mathcal{N}(0, \mathcal{R})$ noise.

C. A Bayesian way of optimizing regularization

One of the main questions is how to chose the regularization matrix W_r, such that an optimal bias-variance trade-off is found. A recently introduced computationally efficient data-driven approach follows an *empirical Bayes method* [19]. It is assumed that the parameter vector θ_0 has a *Gaussian* prior:

$$\theta_0 \sim \mathcal{N}(\theta_a, P_\alpha), \qquad \theta_a = 0,$$

where the covariance matrix P_α is a function of some hyper parameters $\alpha \in \mathbb{R}_+^{n_\alpha}$. In the Bayesian setting, under the assumption that u and p are given realizations, hence, Φ_N is deterministic, the output vector Y_N and the parameters θ_0 according to (11) are jointly Gaussian variables:

$$\begin{bmatrix} \theta_0 \\ Y_N \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} P_\alpha & P_\alpha\Phi_N \\ \Phi_N^\top P_\alpha & \Phi_N^\top P_\alpha\Phi_N + I_M \otimes \mathcal{R} \end{bmatrix} \right). \quad (15)$$

It can be shown that the maximum *posteriori* estimate and minimal variance estimate of θ_0 given Y_N is equivalent to the weighted regularized least squares estimate $\hat{\theta}_{\text{RWLS}}$, e.g., see [17], if the weighting and regularization matrices are

$$W_e = I_M \otimes \mathcal{R}^{-1}, \qquad W_r = P_\alpha^{-1}. \quad (16)$$

The Gaussian prior makes it possible to create an optimal estimate of $\mathcal{R}$ and α from data. The covariance matrix P_α and the noise covariance matrix $\mathcal{R}$ satisfy

$$Y_N \sim \mathcal{N}\left(0, \Phi_N^\top P_\alpha\Phi_N + I_M \otimes \mathcal{R} \right). \quad (17)$$

Hence, the likelihood function of the observation Y_N given α and $\mathcal{R}$ can be used to arrive to their posteriori estimate:

$$\hat{\alpha} = \underset{\alpha}{\text{argmax}} \;\; f(Y_N|\alpha) = \underset{\alpha}{\text{argmin}} \;\; -2\log f(Y_N|\alpha)$$
$$= \underset{\alpha}{\text{argmin}} \;\; \log\left(\det\left(\Phi_N^\top P_\alpha\Phi_N + I_M \otimes \mathcal{R} \right) \right)$$
$$+ Y_N^\top \left(\Phi_N^\top P_\alpha\Phi_N + I_M \otimes \mathcal{R} \right)^{-1} Y_N, \quad (18)$$

where the constant terms are excluded and $f(\cdot)$ is the probability density function of the multivariate normal distribution. In general, $\mathcal{R}$ can also be estimated by the ML optimization or by first estimating an FIR or ARX model with least squares and use the sample variance as the estimate of the noise variance, e.g., see [13]. However, due to space limitation, in this paper $\mathcal{R}$ is assumed to be known. For

a detailed description of pros and cons of the empirical Bayes method compared to other methods, see [14], [17]. The choice of the parametrization of P_α is of high importance as it governs the "quality" of the estimate. For this purpose, commonly different kernel functions are employed, see [17] for a detailed discussion. In the Appendix, a novel extension of [13] to the MIMO case is given, which is a numerically attractive and robust method to maximize the ML (18) and to compute the estimate (14) by QR factorizations.

IV. LPV State-Space Realization

After identification of the sub-Markov parameters by the Bayesian procedure described in Sec. III, the next step of the identification scheme is to realize the LPV-SS model, which is performed by the *basis reduced Ho-Kalman* realization algorithm detailed in [16]. The estimated sub-Markov parameters form the sub-Hankel matrices [16, Eq. 20], whereupon the LPV-SS model is realized. The realization scheme can also be used to efficiently determine the model order. The key advantage of this scheme is that only the non-repetitive elements of the "extended" Hankel matrix are selected, which significantly reduces the matrix on which the SVD is computed from $n_y \sum_{l=1}^{i}(1+n_\psi)^i \times n_u \sum_{l=1}^{j}(1+n_\psi)^j$ with $i, j \geq 2$ to $n_o \times n_r$ with $n_o, n_r \geq n_x$. The performance of the scheme is demonstrated in Sec. VI by identifying a real-world sized system within 2sec on a standard laptop. Due to space limitations, the underlying derivations are not given.

V. Refinement via Expectation Maximization

After executing the bases reduced Ho-Kalman realization scheme an estimate $\hat{\Omega}$ of Ω_0 is found. However, even if the FIR estimation approach of Sec. III delivers an ML estimate and the realization step is exact, we do not know how the truncation n_h effects $\hat{\Omega}$ and how the nonlinear projection of the sub-Markov parameters to $\hat{\Omega}$ effects their variance. Hence, to reach an ML estimate of Ω, a refinement step is required. For this purpose, we propose to use the EM method as Step 3 of the identification scheme, now under the assumption that both v_x and v_y are non-zero random processes as (2). This method also estimates the unknown noise covariance Σ_v and the initial state

$$x_0 \sim \mathcal{N}(\mu, P_0). \qquad (19)$$

The overall parametrization is given as

$$\Gamma = \begin{bmatrix} \text{vec}(\Omega^\top)^\top & \mu^\top & \text{vec}(\Sigma_v)^\top & \text{vec}(P_0)^\top \end{bmatrix}^\top. \qquad (20)$$

The EM method is an iterative optimization approach, which requires an efficient initialization $\hat{\Gamma}_0$, else it might not converge. The previous steps provide an efficient pre-estimate of $\hat{\Omega}$, which is often "close" to the target ML estimate. The aim is to reconstruct the state sequence x and maximize the associated marginal likelihood given the current estimate $\hat{\Gamma}_k$. The main reasoning and a brief overview is given in this section, see [12] for details.

A. The principal idea

Employing Bayes' rule on $f_\Gamma(x_{t+1}, y_t)$ gives

$$\log f_\Gamma(y_t) = \log f_\Gamma(x_{t+1}, y_t) - \log f_\Gamma(x_{t+1}|y_t), \qquad (21)$$

where $f_\Gamma(\mathbf{x}|\mathbf{y})$ is a shorthand for the probability density function of $\mathbf{x}$ conditional on $\mathbf{y}$. Note that maximizing $f_\Gamma(x_{t+1}, y_t)$ w.r.t. Γ is a linear-in-the-parameter problem given the model (1) with dependency structure (3) and noise structure (2), as highlighted later. The state sequence can only be reconstructed by an estimate $\hat{\Gamma}$ of Γ. To achieve this, based on $\mathcal{D}_N$, the probability functions $f_\Gamma(x_{t+1}, y_t)$, $f_\Gamma(x_{t+1}|y_t)$ are approximated by their expectation for an estimate $\hat{\Gamma}$, i.e.,

$$Z(\Gamma) = \mathbb{E}_{\hat{\Gamma}}\{\log f_\Gamma(x_{t+1}, y_t)|\mathcal{D}_N\} - \mathbb{E}_{\hat{\Gamma}}\{\log f_\Gamma(x_{t+1}|y_t)|\mathcal{D}_N\}$$
$$= \mathcal{Q}(\Gamma, \hat{\Gamma}) - \mathcal{V}(\Gamma, \hat{\Gamma}), \qquad (22)$$

where $Z(\Gamma) \triangleq \log f_\Gamma(y_t)$. The term $\mathcal{V}(\hat{\Gamma}, \hat{\Gamma}) - \mathcal{V}(\Gamma, \hat{\Gamma}) \geq 0$ is the *Kullback-Leibler divergence metric* between $\log f_\Gamma(x_{t+1}|y_t)$ and $\log f_{\hat{\Gamma}}(x_{t+1}|y_t)$, which is minimal for $\hat{\Gamma} = \Gamma_0$. Hence,

$$\mathcal{Q}(\Gamma, \hat{\Gamma}) > \mathcal{Q}(\hat{\Gamma}, \hat{\Gamma}) \Rightarrow Z(\Gamma) > Z(\hat{\Gamma}). \qquad (23)$$

So, any new Γ which increases $\mathcal{Q}(\Gamma, \hat{\Gamma})$ compared to $\mathcal{Q}(\hat{\Gamma}, \hat{\Gamma})$, also increases its corresponding log-likelihood value. This principle is used to refine an estimate $\hat{\Gamma}_k$ iteratively by the following two steps:

1) Expectation step, which calculates $\mathcal{Q}(\Gamma, \hat{\Gamma}_k)$.
2) Maximization step, which computes

$$\hat{\Gamma}_{k+1} = \underset{\Gamma}{\text{argmax}} \ \mathcal{Q}(\Gamma, \hat{\Gamma}_k). \qquad (24)$$

B. The expectation step

According to our setting in Sec.II-A and Bayes' rule, the state and the output is distributed as follows

$$f_\Gamma\left(\begin{bmatrix} x_{t+1} \\ y_t \end{bmatrix} \middle| x_t, u_t\right) \sim \mathcal{N}\left(\Gamma\begin{bmatrix} \psi_t \otimes x_t \\ \psi_t \otimes u_t \end{bmatrix}, \Sigma_v\right). \qquad (25)$$

Then, the approximated log-likelihood function $\mathcal{Q}(\Gamma, \hat{\Gamma}_k)$ is:

Lemma 2: Based on the LPV-SS model structure (1), parametrization (3), and Gaussian prior (19) on the initial state, the joint log-likelihood approximation $\mathcal{Q}(\Gamma, \hat{\Gamma}_k)$ is

$$-2\mathcal{Q}(\Gamma, \hat{\Gamma}_k) = \log \det P_1 + N \log \det \Sigma_v +$$
$$\text{Tr}\left(P_1^{-1}\mathbb{E}_{\hat{\Gamma}_k}\left\{(x_1 - \mu)(x_1 - \mu)^\top \mid y\right\}\right) +$$
$$\text{Tr}\left(\Sigma_v^{-1}\left[\Phi - \Psi\Gamma^\top - \Gamma\Psi^\top + \Gamma\Sigma\Gamma^\top\right]\right), \qquad (26)$$

where

$$\Phi = \sum_{t=1}^{N} \begin{bmatrix} \mathbb{E}_{\hat{\Gamma}_k}\{x_{t+1}x_{t+1}^\top|\mathcal{D}_N\} & \hat{x}_{t+1|N}y_t^\top \\ y_t\hat{x}_{t+1|N}^\top & y_ty_t^\top \end{bmatrix}, \qquad (27a)$$

$$\Psi = \sum_{t=1}^{N} \begin{bmatrix} \psi_t^\top \otimes \mathbb{E}_{\hat{\Gamma}_k}\{x_{t+1}x_t^\top|\mathcal{D}_N\} & \psi_t^\top \otimes \hat{x}_{t+1|N}u_t^\top \\ \psi_t^\top \otimes y_t\hat{x}_{t|N}^\top & \psi_t^\top \otimes y_tu_t^\top \end{bmatrix}, \qquad (27b)$$

$$\Sigma = \sum_{t=1}^{N} \begin{bmatrix} \psi_t\psi_t^\top \otimes \mathbb{E}_{\hat{\Gamma}_k}\{x_tx_t^\top|\mathcal{D}_N\} & \psi_t\psi_t^\top \otimes \hat{x}_{t|N}u_t^\top \\ \psi_t\psi_t^\top \otimes u_t\hat{x}_{t|N}^\top & \psi_t\psi_t^\top \otimes u_tu_t^\top \end{bmatrix}, \qquad (27c)$$

$$\hat{x}_{t|N} = \mathbb{E}_{\hat{\Gamma}_k}\{x_t|\mathcal{D}_N\}. \qquad (27d)$$

□

Proof: The joint density function $f_\Gamma(x_{t+1}, y_t)$ is a product of density functions for each time instance, which

can be shown by applying Bayes' rule on the Markov process given by the pdf (25). Proving (26) directly follows using (25) with a given $\hat{\Gamma}_k$, see, e.g., [12, Lemma 1]. ∎

The components $\mathbb{E}_{\hat{\Gamma}_k}\{x_t | \mathcal{D}_N\}$, $\mathbb{E}_{\hat{\Gamma}_k}\{x_t x_t^\top | \mathcal{D}_N\}$, and $\mathbb{E}_{\hat{\Gamma}_k}\{x_t x_{t-1}^\top | \mathcal{D}_N\}$ are estimated in three steps, first by applying forward Kalman filtering, then using a backward Kalman smoother, and last, a backward one-lag covariance smoother. Moreover, the required quantities are determined by a *square-root* Kalman filtering strategy, guaranteeing that the state-covariance matrices will be semi-positive definite and symmetric, even for finite precision computations. However, these computations will not be presented here and the interested reader is referred to [12, Lemma 2].

C. The maximization step

With an estimate of the state trajectory, the marginal likelihood maximization has an analytic solution:

Lemma 3: Based on (27) with a positive definite Σ_v, the estimate of Γ is computed as

$$\hat{\Omega} = \Psi\Sigma^{-1}, \qquad \hat{\mu} = \hat{x}_{0|N},$$
$$\hat{\Sigma}_v = L_{22}^\top L_{22}, \qquad \hat{P}_0 = P_{0|N}^{1/2} P_{0|N}^{\top/2}, \qquad (28)$$

where $P_{0|N}$ is the smoothed state-covariance matrix and L_{22} is defined by the upper Cholesky factorization

$$\begin{bmatrix} \Sigma & \Psi^\top \\ \Psi & \Phi \end{bmatrix} = \begin{bmatrix} L_{11}^\top & 0 \\ L_{12}^\top & L_{22}^\top \end{bmatrix} \begin{bmatrix} L_{11} & L_{12} \\ 0 & L_{22} \end{bmatrix}. \qquad (29)$$

Proof: See [12, Lemma 3]. □

Factorization (29) also ensures that $\hat{\Sigma}_v$ is symmetric and positive definite. The EM method described in this section has a low computational load due to its analytic form and the method is more robust to initialization compared to PEM gradient-based methods [11]. However, the EM method still needs an efficient initial estimate and its convergence rate is much lower near the optimum compared to its gradient-based counterparts. Hence, the previous steps of the identification scheme is our contribution to provide a computational efficient initial estimate to avoid possible local minima, as will be shown empirically in Sec. VI. In addition, the EM method iteratively maximizes the ML with its maximum at $\hat{\Gamma} = \Gamma_0$, therefore, if the method converges and $N \to \infty$ then $\hat{\Gamma} \to \Gamma_0$, giving a consistent estimate $\hat{\Omega}$ of Ω.

VI. SIMULATION EXAMPLE

In this section, the performance of the developed 3-step identification procedure is assessed on a Monte-Carlo based simulation example using a randomly generated stable LPV-SS model. The Monte-Carlo study shows the performance of the method in the following cases:

C1 Bayesian FIR estimation with basis reduced Ho-Kalman LPV-SS realization (without the EM refinement step),

C2 Bayesian FIR estimation with basis reduced Ho-Kalman LPV-SS realization and EM refinement step.

The two cases are used to assess if the initialization is efficient and how much additional performance gain can be obtained by the refinement step. The case study is performed on a Macbook pro laptop, late 2013 with 2.6GHz Intel core i5 and Matlab 2014b.

A. Data-generating system and model structure

The data-generating system is randomly generated in terms of an SS model (1) with input-output dimensions $n_{\mathrm{u}} = n_{\mathrm{y}} = 2$, scheduling dimension $n_\psi = 5$, minimal state dimension $n_{\mathrm{x}} = 4$, and affine dependence, i.e., the known basis functions are $\psi^{[i]} = p^{[i]}$ with $p^{[i]}$ denoting the ith element of p. The SS model represented system is asymptotically quadratically stable on the domain $p_t \in \mathbb{P} = [-1,1]^5$. Note that this model already borders the computational limits of other LPV state-space approaches [8, Table 1]. The LPV-SS model is available at www.rolandtoth.us.

B. Identification setting

The identification dataset is constructed from a white u with uniform distribution $u_t \sim \mathcal{U}(-1,1)$, and white p with random binary distribution on $\{-0.9, 0.9\}$, each of length $N = 5 \cdot 10^3$. v_{x} is considered to be zero, while v_{y} is a white noise with distribution $v_{\mathrm{y}} \sim \mathcal{N}(0, \mathcal{R})$ where $\mathcal{R}$ is diagonal and it is chosen such that the *signal-to-noise ratio* (SNR)

$$\mathrm{SNR}_{\mathrm{y}}^{[i]} = 10\log\frac{\sum_{t=1}^N (y_t^{[i]})^2}{\sum_{t=1}^N (w_t^{[i]})^2},$$

is set for various Monte-Carlo experiments as $\mathrm{SNR}_{\mathrm{y}}^{[i]} = \{40, 20, 10, 0\}$dB for all $i = 1, \ldots, n_{\mathrm{y}}$. The $[i]$ denotes the i-th channel, i.e., element of the vector signals, and $\mathrm{SNR}_{\mathrm{y}}^{[i]}$ is the corresponding SNR on the output $y^{[i]}$. The quality of the estimated models is tested on a noiseless validation data set $\mathcal{D}_{\mathrm{val}}$ of length $N_{\mathrm{val}} = 200$, with different excitation conditions as the estimation data set in terms of

$$u(t) = \begin{bmatrix} 0.5\cos(0.035t) \\ 0.5\sin(0.035t) \end{bmatrix} + \delta_u(t), \qquad (30)$$

$$p^{[i]}(t) = 0.25 - 0.05i + 0.4\sin\left(0.035t + \frac{2i\pi}{5}\right) + \delta_{p_i}(t), \quad (31)$$

where $\delta_u(t) \in \mathbb{R}^{n_{\mathrm{u}}}$, $\delta_{p_i}(t) \in \mathbb{R}$ are element wise i.i.d. sequences with $\mathcal{U}(-0.15, 0.15)$. To study the statistical properties of the developed identification scheme, a Monte-Carlo study with $N_{\mathrm{MC}} = 100$ runs is carried out, where in each run a new realization of the input, scheduling, and noise sequences are taken. The simulated output $\hat{y}$ of the estimated model w.r.t. $\mathcal{D}_{\mathrm{val}}$ is compared to the true output y of the data-generating system by means of the *best fit rate* (BFR) and the *variance accounted for* (VAF)[3]

$$\mathrm{BFR} = \max\left\{1 - \frac{\frac{1}{N}\sum_{t=1}^N \|y_t - \hat{y}_t\|_2}{\frac{1}{N}\sum_{t=1}^N \|y_t - \bar{y}\|_2}, 0\right\} \cdot 100\%, \qquad (32a)$$

$$\mathrm{VAF} = \max\left\{1 - \frac{\frac{1}{N}\sum_{t=1}^N \|y_t - \hat{y}_t - \bar{e}\|_2^2}{\frac{1}{N}\sum_{t=1}^N \|y_t - \bar{y}\|_2^2}, 0\right\} \cdot 100\%. \qquad (32b)$$

In (32a), $\bar{y}$ defines the mean of the true noiseless output y_t in $\mathcal{D}_{\mathrm{val}}$, $\hat{y}_t$ is the simulated output w.r.t. (30) and (31) in $\mathcal{D}_{\mathrm{val}}$, and $\bar{e}$ in (32b) is the mean of the error between y_t and $\hat{y}_t$. As the Ho-Kalman like realization scheme is deterministic, i.e., no noise process is identified, only the

---

[3]Usually the BFR and VAF are defined per channel. Eq. (32a)-(32b) are the average performance criteria over all channels.

TABLE I: Mean and standard deviation (std) of the BFR, VAF, and execution time of the identification algorithm per Monte-Carlo run for different $\mathrm{SNR_y} = \{\infty, 40, 25, 10, 0\}$dB. The performance criteria are based on the simulated output of the estimated model w.r.t. $\mathcal{D}_{\mathrm{val}}$ and $N_{\mathrm{MC}} = 100$ Monte-Carlo simulations. Simulations are performed on a Macbook pro laptop, late 2013 with 2.6GHz and Matlab 2014b.

		C1					**C2**				
		∞ dB	40dB	25dB	10dB	0dB	∞ dB	40dB	25dB	10dB	0dB
BFR [%]	mean	97.09	96.96	97.01	94.17	82.71	99.87	99.81	99.18	95.43	80.07
	std	0.7281	0.8001	0.6909	1.328	4.357	$3.546 \cdot 10^{-2}$	$5.103 \cdot 10^{-2}$	0.1857	1.174	4.291
VAF [%]	mean	99.93	99.92	99.92	99.70	97.37	99.99	99.99	99.99	99.82	96.96
	std	$3.756 \cdot 10^{-2}$	$5.230 \cdot 10^{-2}$	$3.810 \cdot 10^{-2}$	0.1474	1.298	$9.500 \cdot 10^{-5}$	$1.900 \cdot 10^{-4}$	$3.231 \cdot 10^{-3}$	0.1000	1.198
Time Elapsed [s]	mean	1.814	1.820	1.737	1.714	1.700	6.605	6.487	6.373	6.500	6.359
	std	0.1797	0.1465	$7.051 \cdot 10^{-2}$	0.0998	$7.918 \cdot 10^{-2}$	0.2639	0.2667	0.1884	0.2241	0.1339

simulation error will be assessed for the BFR and VAF. In the realization step, the basis reduced scheme uses $n_{\mathrm{o}} = n_{\mathrm{r}} = 10$ bases, where the controllability matrix is spanned by $\varsigma = \{(\varepsilon, 0, 1), (\varepsilon, 1, 1), (\varepsilon, 1, 2), (\varepsilon, 2, 1), (\varepsilon, 3, 1), \ldots, (\varepsilon, 5, 2)\}$ and the observability is spanned by $\nu = \{(1, 0, \varepsilon), (2, 0, \varepsilon), \ldots, (2, 4, \varepsilon)\}$. The truncation order is $n_{\mathrm{h}} = 2$ with $n_\theta = 1032$.

C. Analysis of the results

Table I shows the mean and standard deviation of the BFR and VAF of the simulation error on $\mathcal{D}_{\mathrm{val}}$ and the average time elapsed for $N_{\mathrm{MC}} = 100$ Monte-Carlo runs. All performance criteria indicate that the additional refinement step, with the expectation maximization method, will lead to a better estimate of the model, as expected. Only in case of the $\mathrm{SNR}_{\mathrm{y}}^{[i]} = 0$dB noise scenario, the refinement step will not improve the estimate. In this case, the EM method is not able to converge due to the large noise contribution. However, from the performance criteria it is evident that **C1** by itself leads to an accurate model estimate, hence, the initial estimate is quite accurate under such a noise in itself.

Fig. 1 and 2 show the autocorrelation of the residuals with 99% confidence bound and the cross-correlation between the residuals and the input, see [20, Sec. 16.6]. The figures are based upon the one-step-ahead predictor on the validation dataset where, in addition, an additive $\mathrm{SNR}_{\mathrm{y}}^{[i]} = 25$dB white Gaussian output noise is added. Based upon the cross-correlation, it can be seen that the process models are not invalidated. However, the autocorrelation indicates that the predictor model for **C1** (Fig. 1) can be more accurate and the EM method can be successfully used for refinement of the estimates. The obtained results show that the estimation, realization, and refinement methods presented are capable of identifying the original LPV system even under SNR of 0dB and large scheduling dimensions.

VII. CONCLUSION

In this paper a computationally attractive 3-step LPV-SS identification and realization scheme is presented from a Bayesian perspective. The proposed scheme can identify real-world sized MIMO LPV-SS models under harsh noise conditions within seconds on a regular laptop. In Step 1, the Bayesian truncated IIR estimation is a least squares problem where the regularization parameters are tuned using an empirical Bayesian approach, to avoid the need for a second dataset. The least squares problem is unbiased, even in the presence of a colored output noise. In Step 2, the dimensions of the sub-Hankel matrices in the basis reduced Ho-Kalman

realization scheme are chosen by the user and, thus, the computational complexity. Hence, the combination of Step 1 and Step 2 result in a computationally efficient initial LPV-SS estimate. In Step 3, to reach the ML estimate, the initial LPV-SS model is refined by applying the EM method. The EM method is relatively straightforward, it appears to be robust to initial parameter estimates, and the computational load scales linearly with the dataset length. Finally, to increase the accuracy of the initial LPV-SS estimate even further, this work is intended to be extended to include more involved parametrizations on the correlation matrix P_α.

APPENDIX
NUMERICAL IMPLEMENTATION

To numerically minimize (18), it is important to compute, for a given α, the log likelihood (18) efficiently. However, (18) contains inversions and determinants of large matrices. In [13], a numerically attractive approach is given to compute the likelihood, which is here extended to the MIMO case. As the correlation matrices P_α and $\mathcal{R}$ are positive definite, for a given α, take their lower Cholesky factorization

$$P_\alpha = L_1 L_1^\top, \qquad \mathcal{R} = L_2 L_2^\top, \qquad (33)$$

and use this factorization to define

$$\tilde{Y}_N = \left(\mathrm{I}_M \otimes L_2^{-1}\right) Y_N, \qquad \tilde{\Phi}_N = \Phi_N \left(\mathrm{I}_M \otimes \left(L_2^\top\right)^{-1}\right). \quad (34)$$

By first applying Sylvester's determinant theorem and the Schur complement to the elements within the $\det(\cdot)$ in (18), then using the matrix inversion Lemma on the remaining part of (18), substitution by (33)-(34) allows to rewrite the log likelihood (18) as

$$
\begin{aligned}
\hat{\alpha} = \underset{\alpha}{\arg\min} \ &N \log\left(\det(L_2)^2\right) \\
&- \tilde{Y}_N^\top \tilde{\Phi}_N^\top L_1 \left(\mathrm{I}_{n_\theta} + L_1^\top \tilde{\Phi}_N \tilde{\Phi}_N^\top L_1\right)^{-1} L_1^\top \tilde{\Phi}_N \tilde{Y}_N \\
&+ \log\left(\det\left(\mathrm{I}_{n_\theta} + L_1^\top \tilde{\Phi}_N \tilde{\Phi}_N^\top L_1\right)\right) + \tilde{Y}_N^\top \tilde{Y}_N. \quad (35)
\end{aligned}
$$

Note that the dimensions changed from $n_{\mathrm{y}} M \times n_{\mathrm{y}} M$ to $n_\theta \times n_\theta$, which decreases the computational complexity, as commonly $n_{\mathrm{y}} M \gg n_\theta$. Unfortunately, $\mathrm{I}_{n_\theta} + L_1^\top \tilde{\Phi}_N \tilde{\Phi}_N^\top L_1$ is often ill-conditioned, hence, for numerical reliability, QR factorization is used to compute the marginal likelihood cost in (35) via a two-step approach. First, lets assume, without loss of generality, that

$$\mathrm{rank}\begin{bmatrix} \tilde{\Phi}_N^\top & Y_N \end{bmatrix} = n_\theta + 1, \qquad (36)$$

and that $\begin{bmatrix} \tilde{\Phi}_N^\top & Y_N \end{bmatrix}$ has a unique thin QR factorization, see [21, Theorem 5.2.3, p. 248], which corresponds to the

fact that $I_{n_\theta} + L_1^\top \tilde{\Phi}_N \tilde{\Phi}_N^\top L_1$ is positive definite, hence (35) has a unique solution. The two steps are:

1) pre-processing step: take the thin QR-factorization

$$\begin{bmatrix} \tilde{\Phi}_N^\top & \check{Y}_N \end{bmatrix} = Q_1 R_1 = Q_1 \begin{bmatrix} R_{1,1} & R_{1,2} \end{bmatrix}, \qquad (37)$$

where $Q_1 \in \mathbb{R}^{n_y M \times n_\theta + 1}$ with $Q_1^\top Q_1 = I_{n_\theta + 1}$, $R_{1,1} \in \mathbb{R}^{n_\theta + 1 \times n_\theta}$, and $R_{1,2} \in \mathbb{R}^{n_\theta + 1 \times 1}$.

2) cost function: take the thin QR-factorization

$$\begin{bmatrix} R_{1,1} L_1 & R_{1,2} \\ I_{n_\theta} & 0 \end{bmatrix} = Q_2 R_2 = Q_2 \begin{bmatrix} R_{2,1} & R_{2,2} \\ 0 & r \end{bmatrix}, \qquad (38)$$

where $Q_2 \in \mathbb{R}^{2n_\theta + 1 \times n_\theta + 1}$ has the property $Q_2^\top Q_2 = I_{n_\theta + 1}$, $R_{2,1} \in \mathbb{R}^{n_\theta \times n_\theta}$, $R_{2,2} \in \mathbb{R}^{n_\theta \times 1}$, and $r \in \mathbb{R}$. Combining (37) and (38) gives

$$\begin{bmatrix} \tilde{\Phi}_N^\top L_1 & \check{Y}_N \\ I_{n_\theta} & 0 \end{bmatrix} = \begin{bmatrix} Q_1 & 0 \\ 0 & I_{n_\theta} \end{bmatrix} Q_2 R_2. \qquad (39)$$

Using (39) and the orthogonality property of Q provides

$$R_{2,1}^\top R_{2,1} = L_1^\top \tilde{\Phi}_N \tilde{\Phi}_N^\top L_1 + I_{n_\theta}, \quad R_{2,2}^\top R_{2,2} + r^2 = \check{Y}_N^\top \check{Y}_N, \\ R_{2,2}^\top R_{2,1} = \check{Y}_N^\top \tilde{\Phi}_N^\top L_1, \qquad (40)$$

which simplifies the marginal likelihood (35) as follows

$$\hat{\alpha} = \underset{\alpha}{\arg\min} \ N \log \left(\det(L_2)^2 \right) + \log \left(\det(R_{2,1})^2 \right) + r^2, \quad (41)$$

where $R_{2,1}$ and r^2 are depended on the hyper-parameters α. The cost function in (41) is computed for each iteration and (41) is evaluated in a standard line-search algorithm, e.g., `fmincon` in Matlab. Note that the pre-processing step is used to decrease the dimension from $(n_y M + n_\theta) \times (n_\theta + 1)$ to $(2n_\theta + 1) \times (n_\theta + 1)$ of the QR factorization (38) in each iteration. Furthermore, Eq. (41) avoids any inversion of matrices as seen in (18). In addition, QR-factorization of $\hat{\alpha}$ can be used to compute the parameter estimate in (14) by

$$\hat{\theta}_{\text{RWLS}} = L_1 R_{2,1}^{-1} R_{2,2}. \qquad (42)$$

REFERENCES

[1] P. Lopes dos Santos, T.-P. Azevedo-Perdicoúlis, C. Novara, and D. E. R. J. A. Ramos, Eds., *Linear parameter-varying system identification: new developments and trends*. World Scientific, 2011.

[2] R. Tóth, *Modeling and identification of linear parameter-varying systems*. Springer, 2010.

[3] J. Mohammadpour and C. Scherer, Eds., *Control of linear parameter varying systems with applications*. Springer, 2012.

[4] T. Luspay, B. Kulcsár, J. W. Van Wingerden, and M. Verhaegen, "On the identification of LPV traffic flow model," in *Proc. of the European Control Conf.*, Budapest, Hungary, Aug. 2009, pp. 1752–1757.

[5] A. A. Bachnas, R. Tóth, A. Mesbah, and J. H. A. Ludlage, "A review on data-driven linear parameter-varying modeling approaches: A high-purity distillation column case study," *J. of Process Control*, vol. 24, no. 4, pp. 272–285, 2014.
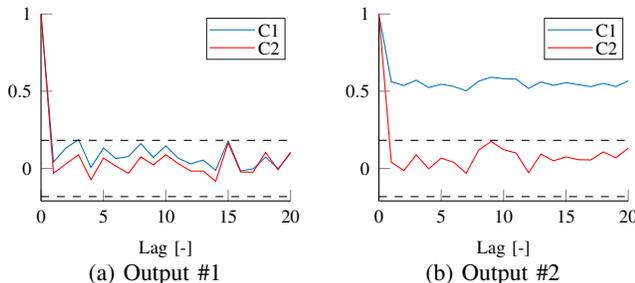
[6] F. Felici, J. W. van Wingerden, and M. Verhaegen, "Subspace identification of MIMO LPV systems using a periodic scheduling sequence," *Automatica*, vol. 43, no. 10, pp. 1684–1697, 2006.

[7] W. E. Larimore, "Identification of nonlinear parameter-varying systems via canonical variate analysis," in *Proc. of the American Control Conf.*, Washington, DC, USA, Jun 2013, pp. 2247–2262.

[8] J. W. van Wingerden and M. Verhaegen, "Subspace identification of bilinear and LPV systems for open- and closed-loop data," *Automatica*, vol. 45, no. 2, pp. 372–381, 2009.

[9] J. W. van Wingerden, "Control of wind yurbines with 'Smart' rotors: proof of concept & LPV subspace identification," PhD thesis, Delft University of Technology, 2008.

[10] M. Sznaier, C. Mazzaro, and T. Inanc, "An LMI approach to control oriented identification of LPV systems," in *Proc. of the American Control Conf.*, Chicago, IL, USA, Jun. 2000, pp. 3682–3686.

[11] L. H. Lee and K. Poolla, "Identification of linear parameter-varying systems using nonlinear programming," *J. of Dynamic Systems, Measurement, and Control*, vol. 121, no. 1, pp. 71–78, 1999.

[12] A. Wills and B. Ninness, "System identification of linear parameter varying state-space models," in *Linear parameter-varying system identification: new developments and trends*, P. Lopes dos Santos, T. P. Azevedo Perdicoúlis, C. Novara, J. A. Ramos, and D. E. Rivera, Eds. World Scientific, 2011, ch. 11, pp. 295–316.

[13] T. Chen and L. Ljung, "Implementation of algorithms for tuning parameters in regularized least squares problems in system identification," *Automatica*, vol. 49, no. 7, pp. 2213–2220, 2013.

[14] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, no. 3, pp. 657–682, 2014.

[15] M. A. H. Darwish, P. B. Cox, G. Pillonetto, and R. Tóth, "Bayesian identification of LPV Box-Jenkins models," in *Proc. of the 54th IEEE Conf. on Decision and Control*, Osaka, Japan, Dec. 2015, pp. 66–71.

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

[17] T. Chen, H. Ohlsson, and L. Ljung, "On the estimation of transfer functions, regularizations and Gaussian processes," *Automatica*, vol. 48, no. 8, pp. 1525–1535, 2012.

[18] V. Laurain, M. Gilson, R. Tóth, and H. Garnier, "Refined instrumental variable methods for identification of LPV Box-Jenkins models," *Automatica*, vol. 46, no. 6, pp. 959–967, 2010.

[19] B. P. Carlin and T. A. Louis, *Bayes and empirical Bayes methods for data analysis*, 2nd ed. London: Chapman & Hall, 1996.

[20] L. Ljung, *System identification: theory for the user*. Springer, 1989.

[21] G. Golub and C. Van Loan, *Matrix computations*, 4th ed. The Johns Hopkins University Press, 2013.

(a) Output #1 (b) Output #2

Fig. 1: Autocorrelation of the residual for the noisy validation dataset with $\text{SNR}_y = 25$dB and the 99% confidence bound.

(a) Input #1 Output #1 (b) Input #1 Output #2
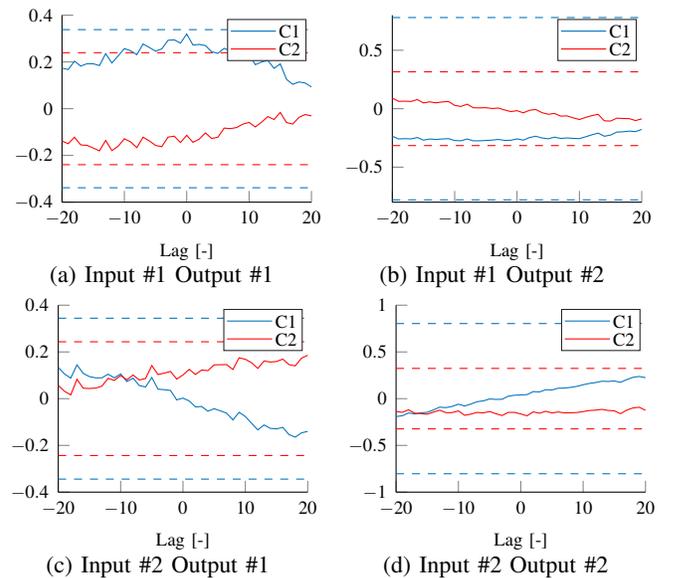
(c) Input #2 Output #1 (d) Input #2 Output #2

Fig. 2: Cross correlation of the residual and the input given with solid lines and the 99% confidence bounds are displayed with dashed lines on a noisy validation data set with $\text{SNR}_y^{[t]} = 25$dB.