Brief paper

State-space LPV model identification using kernelized machine learning[☆]

Syed Zeeshan Rizvi [a,*], Javad Mohammadpour Velni [a], Farshid Abbasi [a], Roland Tóth [b], Nader Meskin [c]

[a] *School of Electrical & Computer Engineering, The University of Georgia, Athens, GA 30602, USA*
[b] *Department of Electrical Engineering, Eindhoven University of Tech., P.O. Box 513, 5600 MB Eindhoven, The Netherlands*
[c] *Department of Electrical Engineering, Qatar University, Doha, Qatar*

ARTICLE INFO

ABSTRACT

This paper presents a nonparametric method for identification of MIMO *linear parameter-varying* (LPV) models in state-space form. The states are first estimated up to a similarity transformation via a nonlinear *canonical correlation analysis* (CCA) operating in a *reproducing kernel Hilbert space* (RKHS). This enables to reconstruct a minimal-dimensional inference between past and future input, output and scheduling variables, making it possible to estimate a state sequence consistent with the data. Once the states are estimated, a *least-squares support vector machine* (LS-SVM)-based identification scheme is formulated, allowing to capture the dependency structure of the matrices of the estimated state-space model on the scheduling variables without requiring an explicit declaration of these often unknown dependencies; instead, it only requires the selection of nonlinear kernel functions and the tuning of the associated hyper-parameters.

1. Introduction

Linear parameter-varying (LPV) model identification has attracted a lot of attention within the system identification community in the recent past. Although a significant progress on the identification of LPV systems with *input–output* (IO) models has been achieved (Bamieh & Giarre, 2002; Tóth, 2010; Tóth, Laurain, Zheng, & Poolla, 2011), identification in an LPV state-space form remains challenging with several open problems. The main streams of LPV control synthesis approaches in the literature are derived from LPV *state-space* (SS) representations. However, the bulk of discrete-time LPV identification and modeling is often carried out under an IO structure. Therefore, a possible approach would be to transform available LPV-IO models to LPV-SS form. However, such a transformation is complicated as the conversion to equivalent SS models often results in dynamic dependence of the state-space matrices on the scheduling variables while approximative "realizations" deform the dynamical relations between the inputs and outputs, often leading to high output errors (Tóth, Abbas, & Werner, 2012). Allowing for such a dynamic dependence increases the complexity of the transformed LPV-SS model thereby making controller synthesis more difficult or even computationally infeasible. It is for this reason that LPV-SS models directly identified from IO data are of prime importance.

Broadly speaking, LPV identification methods can be categorized into parametric and nonparametric methods. In parametric identification of LPV models, the assumption is made that the scheduling dependencies of the model coefficients are known *a priori* (Bamieh & Giarre, 2002). However, in practice, selecting adequate functions to parameterize these dependencies is a non-trivial task where often one tries to include a wide array of basis functions to ensure that the process dynamics are captured. This often leads to over-parametrization of the model coefficients (Laurain, Tóth, Zheng, & Gilson, 2012), causing a large variance in the estimates. On the other hand, an inappropriate selection of these functions causes structural bias (Tóth et al., 2011). Examples of parametric LPV-SS identification include subspace identification methods published in van Wingerden and Verhaegen (2009) and Verdult and Verhaegen (2002). These methods pertain to systems that can be modeled with affine parameter-dependence, and are

usually only suitable for low-dimensional cases. For an overview of other LPV-SS identification schemes, see Tóth (2010). An alternative approach with an attractive bias–variance trade-off is to obtain a nonparametric reconstruction of the scheduling dependencies in LPV models. Kernel-based nonparametric identification techniques have demonstrated encouraging results for LPV-IO models in Abbasi, Mohammadpour, Tóth, and Meskin (2014), Laurain et al. (2012) and Tóth et al. (2011), among others; however, very few fully nonparametric methods for state-space model structures have been reported. A mixed parametric method based on *least-squares support vector machine* (LS-SVM) was proposed recently in dos Santos et al. (2014). In this work, the state matrix *A* is described by a parametric model, while the state-readout matrix *C* is described by a nonparametric one. The problem of selecting basis functions therefore is solved only partially. Additionally, the work dos Santos et al. (2014) focuses only on *single-input single-output* (SISO) LPV-SS models. In our recent work Rizvi, Mohammadpour, Tóth, and Meskin (2015b), we proposed an LS-SVM-based LPV-SS identification method for *multi-input multi-output* (MIMO) systems. Further improvement was presented in Rizvi, Mohammadpour, Tóth, and Meskin (2015a) by incorporating instrumental variables, making the technique robust to the presence of colored noise. A limiting factor in both of these works however, was the assumption of the availability of state measurements, which, most often, is not the case in practical situations. To avoid confusion, it is also noted that in van Wingerden and Verhaegen (2009) and Verdult and Verhaegen (2005), kernel-based methods have been applied for an entirely different purpose, namely to increase computational reliability of regression problems in predictor-based subspace identification with a fixed affine dependency structure of the estimated model.

In this paper, we present an LS-SVM-based nonparametric method for MIMO LPV-SS model identification. The proposed technique works in two steps; first, LS-SVM-based nonlinear *canonical correlation analysis* (CCA) is used to estimate the states of the data-generating system from inputs, outputs, and scheduling variables data. The estimated states are then used with the measured data to identify the state-space matrices of an LPV-SS model of the data-generating system with no assumption made *a priori* on the scheduling dependency structure. The main contribution of this paper lies in the formulation of kernelized CCA and LS-SVM for LPV-SS model identification such that the linearity in the dynamic relation of the model is retained. The paper is arranged as follows. The problem is formulated in Section 2. The use of correlation analysis for the estimation of the states is derived and explained in Section 3. Section 4 details the LS-SVM-based identification algorithm. To demonstrate the capabilities of the developed approach, simulation studies are provided in Section 6. Finally, concluding remarks are made in Section 7.

2. Problem formulation and preliminaries

Consider an LPV system represented by the following discrete-time state-space innovation noise model:

$$x_{k+1} = A(p_k)x_k + B(p_k)u_k + K(p_k)e_k, \tag{1a}$$

$$y_k = C(p_k)x_k + D(p_k)u_k + e_k, \tag{1b}$$

where $k \in \mathbb{Z}$ denotes discrete-time, and $A(p_k) \in \mathbb{R}^{n \times n}$, $B(p_k) \in \mathbb{R}^{n \times n_u}$, $K(p_k) \in \mathbb{R}^{n \times n_y}$, $C(p_k) \in \mathbb{R}^{n_y \times n}$, and $D(p_k) \in \mathbb{R}^{n_y \times n_u}$ are smooth functions of time-varying scheduling variables $p_k \in \mathbb{P} \subset \mathbb{R}^{n_p}$ with $\mathbb{P}$ being compact. Variables $u_k \in \mathbb{R}^{n_u}$, $y_k \in \mathbb{R}^{n_y}$, and $x_k \in \mathbb{R}^n$ represent the inputs, outputs, and states of the system at time k, while $e_k \in \mathbb{R}^{n_y}$ is a stochastic white noise process,

independent of *u*. By substituting $e_k = y_k - C(p_k)x_k - D(p_k)u_k$ in (1a), we can re-write the above set of equations as

$$x_{k+1} = \tilde{A}(p_k)x_k + \tilde{B}(p_k)u_k + K(p_k)y_k, \tag{2a}$$

$$y_k = C(p_k)x_k + D(p_k)u_k + e_k, \tag{2b}$$

where $\tilde{A}(p_k) = A(p_k) - K(p_k)C(p_k)$, and $\tilde{B}(p_k) = B(p_k) - K(p_k)D(p_k)$. Similar to the LTI case, (2) must be asymptotically stable in the deterministic sense (even if asymptotic stability of (1) is not required), otherwise identification of (1) is ill-posed due to the divergence of the variance of the resulting stochastic process. Our aim is to develop a kernelized LS-SVM approach to estimate the functional dependencies of the state-space matrices on the scheduling variables, given only the measurements $\mathcal{D} = \{u_k, y_k, p_k\}_{k=1}^N$, where N is the number of samples (see the equation in Box I).

3. A KCCA-based approach for state estimation

To achieve the aforedescribed objective, first we aim at estimating the state sequence $\{x_k\}_{k=1}^N$ compatible with $\mathcal{D}$.

3.1. Canonical correlation analysis

Canonical correlation analysis (CCA) is a statistical method mainly used to determine linear relations among several variables and had a major role in the development of LTI subspace identification. Given two sets of variables, $u \in \mathbb{R}^{n_u}$ and $y \in \mathbb{R}^{n_y}$, with N samples of each collected in $U \in \mathbb{R}^{N \times n_u}$ and $Y \in \mathbb{R}^{N \times n_y}$, CCA aims at finding vectors $v_j \in \mathbb{R}^{n_u}$ and $w_j \in \mathbb{R}^{n_y}$ to maximize correlation between projected variables Uv_j and Yw_j, also known as variates (Suykens, Gestel, Moor, & Vandewalle, 2002). This leads to the following constrained optimization problem:

$$\max_{v_j, w_j} v_j^\top U^\top Y w_j, \quad \text{s.t.} \quad v_j^\top U^\top U v_j = w_j^\top Y^\top Y w_j = 1.$$

The optimization solved in the dual form leads to a generalized eigenvalue problem. For more details, see Verdult, Suykens, Boets, Goethals, and De Moor (2004). CCA only uses second order information to identify the relation between data sets, an important consequence of which is that CCA and its regularized versions are easily *kernelizable* and can handle nonlinear relationships (De Bie & De Moor, 2003).

3.2. Regularized kernel CCA for LPV state estimation

In our LPV problem setting, we aim to use CCA to find an estimate of the state sequence associated with $\mathcal{D}$. However, we will show that the associated relationship between the inputs and outputs of the data-generating LPV model (1) that defines the so-called state map is a heavily nonlinear dynamic function of p_k. This state map can be captured by modifying the linear CCA and incorporating kernel functions to map the nonlinear dynamic relations into a *reproducing kernel Hilbert space* (RKHS), where classical CCA is applied, resulting in a kernelized CCA method (Bach & Jordan, 2003). The main idea behind this is that the states are the minimal interface between the past and future input, output and scheduling variable data; therefore, the states are expected to be the representative of the past behavior needed to determine the future behavior (Verdult et al., 2004).

Define the past scheduling variables $\bar{p}_k^d \in \mathbb{R}^{dn_p}$ and future scheduling variables $\bar{p}_{k+d}^d \in \mathbb{R}^{dn_p}$ w.r.t. time instant k as

$$\bar{p}_k^d := [p_{k-d}^\top \cdots p_{k-1}^\top]^\top, \tag{4a}$$

$$\bar{p}_{k+d}^d := [p_k^\top \cdots p_{k+d-1}^\top]^\top, \tag{4b}$$

where d denotes the number of past and future samples. Past and future inputs and outputs $\bar{u}_k^d \in \mathbb{R}^{dn_u}$, $\bar{y}_k^d \in \mathbb{R}^{dn_y}$, $\bar{u}_{k+d}^d \in \mathbb{R}^{dn_u}$, and

$$
\begin{bmatrix} y_k \\ y_{k+1} \\ \vdots \\ y_{k+d-1} \end{bmatrix} = \underbrace{\begin{bmatrix} C(p_k) \\ C(p_{k+1})\tilde{A}(p_k) \\ \vdots \\ C(p_{k+d-1})\prod_{l=2}^{d}\tilde{A}(p_{k+d-l}) \end{bmatrix}}_{(\mathcal{O}_f^d \diamond p)(k)} x_k + \underbrace{\begin{bmatrix} D(p_k) & 0 & \cdots & 0 \\ C(p_{k+1})\tilde{B}(p_k) & D(p_{k+1}) & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ C(p_{k+d-1})\prod_{l=2}^{d-1}\tilde{A}(p_{k+d-l})\tilde{B}(p_k) & C(p_{k+d-1})\prod_{l=2}^{d-2}\tilde{A}(p_{k+d-l})\tilde{B}(p_{k+1}) & \cdots & D(p_{k+d-1}) \end{bmatrix}}_{(\mathcal{H}_f^d \diamond p)(k)}
$$

$$
\times \begin{bmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+d-1} \end{bmatrix}
$$

$$
+ \underbrace{\begin{bmatrix} 0 & 0 & \cdots & 0 \\ C(p_{k+1})K(p_k) & 0 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ C(p_{k+d-1})\prod_{l=2}^{d-1}\tilde{A}(p_{k+d-l})K(p_k) & C(p_{k+d-1})\prod_{l=2}^{d-2}\tilde{A}(p_{k+d-l})K(p_{k+1}) & \cdots & 0 \end{bmatrix}}_{(\mathcal{L}_f^d \diamond p)(k)} \begin{bmatrix} y_k \\ y_{k+1} \\ \vdots \\ y_{k+d-1} \end{bmatrix} + \begin{bmatrix} e_k \\ e_{k+1} \\ \vdots \\ e_{k+d-1} \end{bmatrix}. \tag{3}
$$

Box I.

$\bar{y}_{k+d}^d \in \mathbb{R}^{dn_y}$ are defined in a similar way. Further, we also define $\bar{z}_k^d = \begin{bmatrix} \bar{u}_k^d \\ \bar{y}_k^d \end{bmatrix}, \bar{z}_{k+d}^d = \begin{bmatrix} \bar{u}_{k+d}^d \\ \bar{y}_{k+d}^d \end{bmatrix} \in \mathbb{R}^{d(n_u+n_y)}$. The future outputs of (2) can be written in the observability form[1] (3), described compactly as

$$
\bar{y}_{k+d}^d = (\mathcal{O}_f^d \diamond p)(k) \cdot x_k + (\mathcal{H}_f^d \diamond p)(k) \cdot \bar{u}_{k+d}^d \\
+ (\mathcal{L}_f^d \diamond p)(k) \cdot \bar{y}_{k+d}^d + \bar{e}_{k+d}^d, \tag{5}
$$

where $(\mathcal{O}_f^d \diamond p)(k) \in \mathbb{R}^{dn_y \times n}$ is the time-varying d-step forward observability matrix[2] at time k along the scheduling trajectory p, $(\mathcal{H}_f^d \diamond p)(k) \in \mathbb{R}^{dn_y \times dn_u}$ is a forward Toeplitz matrix based on the *Infinite Impulse Response* (IIR) coefficients of (2), and $(\mathcal{L}_f^d \diamond p)(k) \in \mathbb{R}^{dn_y \times dn_y}$ is a lower triangle matrix. Variable $\bar{e}_{k+d}^d$ denotes a segment of the sample path of e_k. Without loss of generality, it is assumed that (2) is *structurally observable* in the deterministic sense.[3]

Definition 3.1. The LPV-SS representation (2) with state-dimension n is called structurally observable, if there exists a scheduling trajectory $p \in \mathbb{P}^{\mathbb{Z}}$, such that the n-step observability matrix $(\mathcal{O}_f^n \diamond p)(k)$ is full (column) rank for all $k \in \mathbb{Z}$.

Let $\mathcal{P} \subseteq \mathbb{P}^{\mathbb{Z}}$ be the set of all scheduling trajectories p such that $\text{rank}\big((\mathcal{O}_f^n \diamond p)(k)\big) = n$ for all $k \in \mathbb{Z}$. Then, in order to guarantee that $\mathcal{O}_f^d$ is injective in (5), it is assumed that d is chosen such that $d \geq n$ and $p \in \mathcal{P}$ in the given data set $\mathcal{D}$. In other words, we assume that the to-be-estimated model representation is observable along the given trajectory of p on all intervals of length d, which corresponds to a *persistency of excitation* (PE) condition on p. Dropping

---

[1] Note that, in (3), we use a left precedence for multiplication, *e.g.*, for $k = d = 4$, $\prod_{l=2}^{d} \tilde{A}(p_{k+d-l}) = \tilde{A}(p_6)\tilde{A}(p_5)\tilde{A}(p_4)$.

[2] The notation $(\mathcal{O}_f^d \diamond p)(k)$ corresponds to the evaluation of $\mathcal{O}_f^d$ along p at time instant k and is used as a shorthand to express the dynamic dependence of the corresponding matrix functions, e.g., $\mathcal{O}_f^d$ at time k depends on the instantaneous and future sample values of the scheduling variables, *i.e.*, $p_k, \ldots, p_{k+d-1}$.

[3] If the assumption of structural observability is not satisfied, then (2) is not state-minimal and under some mild assumptions on the class of functional dependencies of the associated matrix functions, there exists a structurally observable LPV-SS realization of the underlying system with equivalent IO map.

the dynamic dependence argument for notational ease, the state sequence statistics can be given by

$$
x_k = \left(\mathcal{O}_f^d(k)\right)^\dagger \left((I - \mathcal{L}_f^d(k))\bar{y}_{k+d}^d - \mathcal{H}_f^d(k)\bar{u}_{k+d}^d\right) - \left(\mathcal{O}_f^d(k)\right)^\dagger \bar{e}_{k+d}^d, \tag{6}
$$

where $(\cdot)^\dagger$ indicates the Moore–Penrose pseudo-inverse, which exists due to injectivity of the linear map $\mathcal{O}_f^d(k)$ in (9). Using a similar definition of a d-step backward reachability matrix $\mathcal{R}_p^d(k)$ depending on $p_{k-d}, \ldots, p_{k-1}$ and its counterpart $\mathcal{V}_p^d(k)$ with respect to $K(p_k)$, x_k can also be given as

$$
x_k = \underbrace{\left(\prod_{i=1}^{d}\tilde{A}(p_{k-i})\right) x_{k-d}}_{\mathcal{X}_p^d(k)} + \mathcal{R}_p^d(k)\bar{u}_k^d + \mathcal{V}_p^d(k)\bar{y}_k^d, \tag{7}
$$

which in turn can be substituted into (5) to obtain a relation of future and past IO data similar to the data equation used in predictor-based subspace identification:

$$
\bar{y}_{k+d}^d = \mathcal{O}_f^d(k)\mathcal{R}_p^d(k)\bar{u}_k^d + \mathcal{H}_f^d(k)\bar{u}_{k+d}^d + \mathcal{O}_f^d(k)\mathcal{V}_p^d(k)\bar{y}_k^d \\
+ (\mathcal{L}_f^d \diamond p)(k) \cdot \bar{y}_{k+d}^d + \mathcal{O}_f^d(k)\mathcal{X}_p^d(k)x_{k-d} + \bar{e}_{k+d}^d, \tag{8}
$$

where d is chosen such that $\mathcal{X}_p^d(k) \approx 0$ due to the asymptotic stability of (2b). This reveals that the state sequence can be estimated via a CCA between past and future IO data.

To illustrate the concept, one can take (6) and since e is an *independent and identically distributed* (i.i.d) zero-mean process, the expected value of the last term on the right-hand side is zero, giving us, in the conditional mean sense, the following unbiased state estimate

$$
\hat{x}_k = \underbrace{\left(\mathcal{O}_f^d(k)\right)^\dagger \begin{bmatrix} -\mathcal{H}_f^d(k) & I - \mathcal{L}_f^d(k) \end{bmatrix}}_{\varphi_f(\bar{p}_{k+d}^d)} \bar{z}_{k+d}^d. \tag{9}
$$

Note that (9) is a non-minimal variance estimator; however, if we are allowed to change the state basis in terms of the CCA applied on (8), we can then determine a state transformation (see the

discussion of Remark 1) that minimizes this variance contribution by maximizing the correlation between the associated variates. Similarly, choosing d such that $\mathcal{X}_\mathrm{p}^d(k) \approx 0$ in (7), we can arrive at the predictor

$$\hat{x}_k = \underbrace{\begin{bmatrix} \mathcal{R}_\mathrm{p}^d(k) & \mathcal{V}_\mathrm{p}^d(k) \end{bmatrix}}_{\varphi_\mathrm{p}(\bar{p}_k^d)} \bar{z}_k^d. \tag{10}$$

Even if both $\varphi_\mathrm{f}(\bar{p}_{k+d}^d)$ and $\varphi_\mathrm{p}(\bar{p}_k^d)$ are unknown mappings (defined by the to-be-estimated matrix functions), the relations (9) and (10) allow the use of CCA for the estimation of $\hat{x}_k$. Note that maximization of the covariance of the variates can result in estimation of $\hat{x}_k$ on different state basis, which can be seen as a p-dependent state transformation applied on these maps. As these maps at least have polynomial dependence on $\bar{p}_{k+d}^d$ and $\bar{p}_k^d$ even in case of affine dependence of the original matrix functions of (1), a tailor-made kernelized formulation of CCA is required for the underlying estimation problem. To develop this formulation, we define the past and future data sets Φ_p, $\Phi_\mathrm{f} \in \mathbb{R}^{N \times n_\mathrm{G}}$ as

$$\Phi_\mathrm{p} := \begin{bmatrix} \varphi_\mathrm{p}(\bar{p}_1^d)\bar{z}_1^d & \cdots & \varphi_\mathrm{p}(\bar{p}_N^d)\bar{z}_N^d \end{bmatrix}^\top, \tag{11a}$$

$$\Phi_\mathrm{f} := \begin{bmatrix} \varphi_\mathrm{f}(\bar{p}_{1+d}^d)\bar{z}_{1+d}^d & \cdots & \varphi_\mathrm{f}(\bar{p}_{N+d}^d)\bar{z}_{N+d}^d \end{bmatrix}^\top, \tag{11b}$$

where $\varphi_\mathrm{p} : \mathbb{R}^{dn_\mathrm{p}} \to \mathbb{R}^{n_\mathrm{G} \times d(n_\mathrm{u}+n_\mathrm{y})}$ and $\varphi_\mathrm{f} : \mathbb{R}^{dn_\mathrm{p}} \to \mathbb{R}^{n_\mathrm{G} \times d(n_\mathrm{u}+n_\mathrm{y})}$ represent unknown feature maps that, respectively, map the past and future scheduling variables into an RKHS $\mathcal{G}_{\check{k}}$ defined uniquely by a symmetric and positive definite kernel function $\check{k} : \mathbb{R}^{dn_\mathrm{p}} \times \mathbb{R}^{dn_\mathrm{p}} \to \mathbb{R}$ (for details, see Cucker & Smale, 2001); variable n_G represents the dimension of this possibly infinite-dimensional feature space. The kernel function, with arguments in the input space $\mathbb{R}^{dn_\mathrm{p}}$, corresponds to an inner product in the RKHS as

$$\left[\varphi_\mathrm{p}(\bar{p}_k^d)\right]_i^\top \left[\varphi_\mathrm{p}(\bar{p}_j^d)\right]_i = \check{k}\left(\bar{p}_k^d, \bar{p}_j^d\right).$$

Now the CCA problem corresponding to the equivalent representation (2) becomes

$$\max_{v_j, w_j} v_j^\top \Phi_\mathrm{f}^\top \Phi_\mathrm{p} w_j \quad \text{s.t.} \quad v_j^\top \Phi_\mathrm{f}^\top \Phi_\mathrm{f} v_j = w_j^\top \Phi_\mathrm{p}^\top \Phi_\mathrm{p} w_j = 1, \tag{12}$$

where $v_j \in \mathbb{R}^{n_\mathrm{G}}$, $w_j \in \mathbb{R}^{n_\mathrm{G}}$ with $j \in \{1, \dots, N\}$ are directions in the feature space $\mathcal{G}_{\check{k}}$, along which the projections of the future and past data have maximum correlation. The KCCA in terms of (12), though regularized, does not provide a useful canonical correlation of the variables. The Authors of this early version of KCCA, Bach and Jordan, argue in Bach and Jordan (2003) that the geometric interpretation of (12) is equivalent to maximizing the cosine of the angle between subspaces generated by the column spaces of Φ_p and Φ_f. If the non-centered Gram matrices are invertible, as for example, when the data points are distinct and Gaussian kernels are used, then the two column spaces are identical and the angle between them is zero. This results in a canonical correlation estimate that is always equal to one. In other words, (12) does not give a useful canonical correlation (Bach & Jordan, 2003). To overcome this "naive kernelization", different variants of KCCA were later proposed that can provide a good estimator for general kernels. An improved regularized version based on LS-SVM was later introduced in Suykens et al. (2002), the primal version of which, adapted to the case of the LPV-SS model (2), is given as

$$\max_{v_j, w_j} \mathcal{J}(v_j, w_j, s, r) =$$

$$\gamma \sum_{k=1}^N \left(s_k r_k - \nu_\mathrm{f} \frac{1}{2} s_k^2 - \nu_\mathrm{p} \frac{1}{2} r_k^2 \right) - \frac{1}{2} v_j^\top v_j - \frac{1}{2} w_j^\top w_j,$$

$$\text{s.t.} \quad s_k = v_j^\top \varphi_\mathrm{f}(\bar{p}_{k+d}^d)\bar{z}_{k+d}^d, \quad r_k = w_j^\top \varphi_\mathrm{p}(\bar{p}_k^d)\bar{z}_k^d, \tag{13}$$

for $k = 1, \dots, N$, where $\gamma, \nu_\mathrm{f}, \nu_\mathrm{p} \in \mathbb{R}^+$ are positive hyper-parameters needed to be chosen. The main advantage of this improved CCA lies in the introduction of the last two terms in the cost function, which help to regularize v_j, w_j, making sure they do not become arbitrarily large. Writing the above problem in a dual form, we define the *Lagrangian* as given in (14), where $\eta_j = [\eta_j^1 \cdots \eta_j^N]^\top \in \mathbb{R}^N$ and $\kappa_j = [\kappa_j^1 \cdots \kappa_j^N]^\top \in \mathbb{R}^N$ are Lagrange multipliers. The dual form of (13) is obtained via the *Karush–Kuhn–Tucker* (KKT) conditions, i.e., finding the derivatives $\frac{\partial \mathcal{L}}{\partial v_j}, \frac{\partial \mathcal{L}}{\partial w_j}, \frac{\partial \mathcal{L}}{\partial s_k}, \frac{\partial \mathcal{L}}{\partial r_k}, \frac{\partial \mathcal{L}}{\partial \eta_j^k}, \frac{\partial \mathcal{L}}{\partial \kappa_j^k}$ and equating them to zero, which are not shown here due to the space limitations.

$$\mathcal{L}(v_j, w_j, s, r, \eta_j, \kappa_j) = \mathcal{J}(v_j, w_j, s, r) - \sum_{k=1}^N \eta_j^k \left(s_k - v_j^\top \varphi_\mathrm{f}(\bar{p}_{k+d}^d)\bar{z}_{k+d}^d \right)$$

$$- \sum_{k=1}^N \kappa_j^k \left(r_k - w_j^\top \varphi_\mathrm{p}(\bar{p}_k^d)\bar{z}_k^d \right), \tag{14}$$

Using these conditions to eliminate the primal decision variables v_j, w_j, s_k, r_k, and substituting $\lambda_j = 1/\gamma$, the above problem can be simplified to a *regularized* generalized eigenvalue problem as

$$K_\mathrm{pp}\kappa_j = \lambda_j(\nu_\mathrm{f} K_\mathrm{ff} + I)\eta_j, \tag{15a}$$

$$K_\mathrm{ff}\eta_j = \lambda_j(\nu_\mathrm{p} K_\mathrm{pp} + I)\kappa_j, \tag{15b}$$

where $K_\mathrm{ff} = \Phi_\mathrm{f}\Phi_\mathrm{f}^\top$ and $K_\mathrm{pp} = \Phi_\mathrm{p}\Phi_\mathrm{p}^\top$ are kernel matrices that express the inner product of the feature maps in $\mathcal{G}_{\check{k}}$. The elements of the kernel matrices are

$$[K_\mathrm{ff}]_{l,m} = \bar{z}_{l+d}^{d\,\top} \check{k}(\bar{p}_{l+d}^d, \bar{p}_{m+d}^d)\bar{z}_{m+d}^d, \tag{16a}$$

$$[K_\mathrm{pp}]_{l,m} = \bar{z}_l^{d\,\top} \check{k}(\bar{p}_l^d, \bar{p}_m^d)\bar{z}_m^d. \tag{16b}$$

Kernel functions can be chosen from a variety of different functions, including, but not limited to *polynomial kernels* $\check{k}(p_i, p_j) = ((p_i \cdot p_j) + 1)^q$, and *radial basis functions* (RBF) $\check{k}(p_i, p_j) = \exp\left(-\frac{\|p_i - p_j\|_2^2}{\sigma^2}\right)$. Parameters $q \in \mathbb{N}$ and $\sigma \in \mathbb{R}^+$ denote the degree of the polynomial and the spread of the RBF function; these are essentially tuning parameters chosen by the user (Schölkopf & Smola, 2002). By solving (15), one can find the primal decision variables $v_j = \Phi_\mathrm{f}^\top \eta_j$ and $w_j = \Phi_\mathrm{p}^\top \kappa_j$, which are obtained by solving the KKT conditions $\frac{\partial \mathcal{L}}{\partial v_j} = 0$, $\frac{\partial \mathcal{L}}{\partial w_j} = 0$ and obtain an estimate of the state evolution of (1) w.r.t. $\mathcal{D}$.

As $K_\mathrm{pp}, K_\mathrm{ff} \in \mathbb{R}^{N \times N}$, the *regularized* generalized eigenvalue problem (15) can have up to $2N$ different solutions $\eta_j, \kappa_j, j = 1, \dots, 2N$, defining the primal decision variables $v_j = \Phi_\mathrm{f}^\top \eta_j$ and $w_j = \Phi_\mathrm{p}^\top \kappa_j$. Since each of these solutions gives a direction in the feature space correlating past data with the future data, each solution can give us one component, i.e., the jth component, of the state variable that ties the past behavior to the future. Therefore, the estimate of a compatible state vector at time instant k follows using the jth solution to the KCCA problem as

$$\check{x}_k^j = v_j^\top \varphi_\mathrm{f}(\bar{p}_{k+d}^d)\bar{z}_{k+d}^d. \tag{17}$$

Substituting above the earlier solved KKT condition gives $v_j = \Phi_\mathrm{f}^\top \eta_j$, and replacing the feature space dot-product $\varphi_\mathrm{f}^\top(\cdot)\varphi_\mathrm{f}(\cdot)$ with a kernel function $\check{k}(\cdot, \cdot)$, we obtain

$$\check{x}_k^j = \eta_j^\top \begin{bmatrix} \bar{z}_{1+d}^{d\,\top} \check{k}(\bar{p}_{1+d}^d, \bar{p}_{k+d}^d) \\ \vdots \\ \bar{z}_{N+d}^{d\,\top} \check{k}(p_{N+d}^d, \bar{p}_{k+d}^d) \end{bmatrix} \bar{z}_{k+d}^d. \tag{18}$$

Similarly, $w_j = \Phi_\mathrm{p}^\top \kappa_j$ gives the estimated jth component of the state vector at time k as

$$\check{x}_k^j = w_j^\top \varphi_\mathrm{p}(\bar{p}_k^d)\bar{z}_k^d = \kappa_j^\top \begin{bmatrix} \bar{z}_1^{d\,\top} \check{k}(\bar{p}_1^d, \bar{p}_k^d) \\ \vdots \\ \bar{z}_N^{d\,\top} \check{k}(p_N^d, \bar{p}_k^d) \end{bmatrix} \bar{z}_k^d. \tag{19}$$

Remark 1. All $2N$ solutions of the regularized generalized eigenvalue problem (15) in terms of normalized eigenvectors can be analytically calculated via the following economical *singular value decomposition* (SVD)

$$\left[\begin{array}{c|c} \nu_{\mathrm{f}}K_{\mathrm{ff}} + I & 0 \\ \hline 0 & \nu_{\mathrm{p}}K_{\mathrm{pp}} + I \end{array} \right]^{-1} \left[\begin{array}{c|c} 0 & K_{\mathrm{pp}} \\ \hline K_{\mathrm{ff}} & 0 \end{array} \right] = W \Sigma \left[\begin{array}{c} V_1 \\ \hline V_2 \end{array} \right]^{\top} \qquad (20)$$

where Σ is a diagonal matrix containing all non-zero singular values, and the corresponding solutions are $\eta_j = [V_1]_j$ and $\kappa_j = [V_2]_j$ with $[\cdot]_j$ denoting the jth column. An effective dimension of $\check{x}$ can be chosen by only taking into account those $\check{x}_k^j$ which are associated with the $\hat{n}$ most significant singular values. From the stochastic point of view, without regularization, the reconstructed sequences $\check{x}_k^j$ are independent and the magnitude of their autocorrelation reveals their significance in the canonical relationship.

Therefore, given d measurements of inputs, outputs, and scheduling variables, a state sequence $\check{x}_k$, compatible with the system, can be estimated by determining the maximum correlation between $\varphi_{\mathrm{f}}(\bar{p}_{k+d}^d)\bar{z}_{k+d}^d$ and $\varphi_{\mathrm{p}}(\bar{p}_k^d)\bar{z}_k^d$ in the CCA sense. The notion of compatibility corresponds to the fact that the state is estimated up to a linear map or state transformation $T : \mathbb{R}^{dn_{\mathrm{p}}} \to \mathbb{R}^{\hat{n} \times n}$, such that $\check{x}_k = \mathbb{E}\{(T \diamond p)(k) \cdot x_k \mid \mathcal{D}\}$ in the conditional mean sense. The state transformation T can have dynamic dependence on $p_k, \ldots, p_{k-d+1}$ (Tóth et al., 2012) and with $\hat{n} \geq n$ it is injective. Therefore, $\check{x}_k$ is estimated in a state-space basis different from the basis of $\hat{x}_k$. This state-space basis and its dimension is determined by the user using the rank-revealing property of the SVD in (20). This is an established practice in subspace identification for LPV (see Felici, van Wingerden, & Verhaegen, 2007; Verdult & Verhaegen, 2005) and LTI systems (see Overschee & De Moor, 1995). Hence, $\check{x}_k$ corresponds to the estimate of the state sequence of an equivalent realization of (2) as

$$\check{x}_{k+1} = (\tilde{A}_{\mathrm{e}} \diamond p)(k)\check{x}_k + (\tilde{B}_{\mathrm{e}} \diamond p)(k)u_k + (K_{\mathrm{e}} \diamond p)(k)y_k,$$
$$y_k = (C_{\mathrm{e}} \diamond p)(k)\check{x}_k + Du_k + e_k, \qquad (21)$$

where subscript e denotes the estimate and $T\tilde{A} = \tilde{A}_{\mathrm{e}}T$, $T\tilde{B} = \tilde{B}_{\mathrm{e}}$, $TK = K_{\mathrm{e}}$ and $C = C_{\mathrm{e}}T$.

Note that the states associated with an LPV-SS representation of the system summarize all information from the past behavior needed to predict the future behavior. In that sense, the past input and output data $\bar{z}_k^d = \left[\bar{u}_k^{d\top} \; \bar{y}_k^{d\top} \right]^{\top}$ forms a non-minimal state representation of the system. This fact of LTI system theory also holds in the LPV case, i.e., for any finite dimensional LPV-SS representation (1) with up to meromorphic p dependencies, it is possible to give an LPV-SS realization with a state vector $\bar{z}_k^d$ which has an equivalent IO map in an almost everywhere sense (i.e., all compatible trajectories are equal, except a subset of trajectories with measure zero due to possible singularity of the matrix functions). To illustrate this fact, we next show that the future output behavior is a function of past data $\bar{z}_k^d$, $\bar{p}_k^d$, and u_k, p_k plus a contribution of the noise in terms of $\bar{e}_k^d$ and e_k.

Lemma 3.1. *Let (2) be structurally observable and $d \geq n$. Then, there exists a function $f : \mathbb{R}^{n_{\mathrm{f}}} \to \mathbb{R}^{n_y}$ with $n_{\mathrm{f}} = (d + 1)(n_{\mathrm{u}} + n_{\mathrm{p}} + n) + dn_y$ such that for any trajectories $p \in \mathcal{P}$, $u \in (\mathbb{R}^{n_{\mathrm{u}}})^{\mathbb{Z}}$ and $e \in (\mathbb{R}^{n_y})^{\mathbb{Z}}$*

$$y_k = f(u_k, p_k, e_k, \bar{z}_k^d, \bar{p}_k^d, \bar{e}_k^d). \qquad (22)$$

Proof. As (2) is observable along any $p \in \mathcal{P}$, shifting (6) d-samples backward in time is used to substitute x_{k-d} in (7) to arrive at

$$x_k = \mathcal{M}_{\mathrm{p}}^d(k)\left((I - \mathcal{L}_{\mathrm{f}}^d(k-d))\bar{y}_k^d - \mathcal{H}_{\mathrm{f}}^d(k-d)\bar{u}_k^d \right)$$
$$+ \mathcal{R}_{\mathrm{p}}^d(k)\bar{u}_k^d + \mathcal{V}_{\mathrm{p}}^d(k)\bar{y}_k^d - \mathcal{M}_{\mathrm{p}}^d(k)\bar{e}_k^d. \qquad (23)$$

where $\mathcal{M}_{\mathrm{p}}^d(k) = \prod_{i=1}^d \tilde{A}(p_{k-i})\left(\mathcal{O}_{\mathrm{f}}^d(k-d)\right)^{\dagger}$. Using (23) and substituting it into (2b) gives

$$y_k = C(p_k)\left[\mathcal{R}_{\mathrm{p}}^d(k) \quad \mathcal{V}_{\mathrm{p}}^d(k) \right] \bar{z}_k^d + D(p_k)u_k$$
$$- C(p_k)\mathcal{M}_{\mathrm{p}}^d(k)\left[\mathcal{H}_{\mathrm{f}}^d(k-d) \quad \mathcal{L}_{\mathrm{f}}^d(k-d) - I \right] \bar{z}_k^d$$
$$+ e_k - C(p_k)\mathcal{M}_{\mathrm{p}}^d(k)\bar{e}_k^d. \qquad (24)$$

4. Matrix function estimation via LS-SVM

Once we have obtained an estimate $\{\check{x}_k\}_{k=1}^N$ of the state sequence $\{x_k\}_{k=1}^N$, we can form an extended data set $\check{\mathcal{D}} = \{u_k, y_k, \check{x}_k, p_k\}_{k=1}^N$ to estimate the matrix functions in (2). We parameterize our LPV-SS model as[4]

$$\check{x}_{k+1} = W_{\mathrm{x}}\varphi_{\mathrm{x}}^{\top}(p_k) + \varepsilon_k, \qquad (25a)$$
$$y_k = W_{\mathrm{y}}\varphi_{\mathrm{y}}^{\top}(p_k) + \varsigma_k, \qquad (25b)$$

where $\check{x}_k$ is the estimate of x_k, ε_k and ς_k are residual errors on the states and outputs, respectively, $W_{\mathrm{x}} = [W_1 \; W_2 \; W_3] \in \mathbb{R}^{n \times 3n_{\mathrm{H}}}$ and $W_{\mathrm{y}} = [W_4 \; W_5] \in \mathbb{R}^{n_y \times 2n_{\mathrm{H}}}$ are weighting matrices. The functions $\varphi_{\mathrm{x}}^{\top}(p_k) \in \mathbb{R}^{3n_{\mathrm{H}} \times 1}$ and $\varphi_{\mathrm{y}}^{\top}(p_k) \in \mathbb{R}^{2n_{\mathrm{H}} \times 1}$ are defined by

$$\varphi_{\mathrm{x}}^{\top}(p_k) = \left[\left(\Phi_1(p_k)\check{x}_k\right)^{\top} \; \left(\Phi_2(p_k)u_k\right)^{\top} \; \left(\Phi_3(p_k)y_k\right)^{\top} \right]^{\top},$$
$$\varphi_{\mathrm{y}}^{\top}(p_k) = \left[\left(\Phi_4(p_k)\check{x}_k\right)^{\top} \; \left(\Phi_5(p_k)u_k\right)^{\top} \right]^{\top},$$

where $\Phi_1, \Phi_4 : \mathbb{R}^{n_{\mathrm{p}}} \to \mathbb{R}^{n_{\mathrm{H}} \times n}$, $\Phi_2, \Phi_5 : \mathbb{R}^{n_{\mathrm{p}}} \to \mathbb{R}^{n_{\mathrm{H}} \times n_{\mathrm{u}}}$, and $\Phi_3 : \mathbb{R}^{n_{\mathrm{p}}} \to \mathbb{R}^{n_{\mathrm{H}} \times n_y}$ are unknown feature maps that map the scheduling variables to a high dimensional (RKHS) $\mathcal{H}_{\bar{k}}$ defined uniquely by the kernel functions $\bar{k}_i : \mathbb{R}^{n_{\mathrm{p}}} \times \mathbb{R}^{n_{\mathrm{p}}} \to \mathbb{R}$ with $i \in 1, \ldots, 5$. From (25a)–(25b), we can gauge that $\tilde{A}(p_k) \sim W_1\Phi_1(p_k)$, $\tilde{B}(p_k) \sim W_2\Phi_2(p_k)$, $K(p_k) \sim W_3\Phi_3(p_k)$, $C(p_k) \sim W_4\Phi_4(p_k)$, and $D(p_k) = W_5\Phi_5(p_k)$, where $\sim$ stands for an equivalent function under a state transformation. The problem in this paper, therefore, reduces to finding the dependency of $W_i\Phi_i(p_k)$ on p_k given the data $\check{\mathcal{D}}$. To achieve this, we aim to minimize the following cost function:

$$\min_{W_{\mathrm{x}}, W_{\mathrm{y}}, \varepsilon, \varsigma} \mathcal{I}(W_{\mathrm{x}}, W_{\mathrm{y}}, \varepsilon, \varsigma) =$$
$$\frac{1}{2}\left(\|W_{\mathrm{x}}\|_{\mathrm{F}}^2 + \|W_{\mathrm{y}}\|_{\mathrm{F}}^2 + \sum_{k=1}^N \varepsilon_k^{\top}\Gamma\varepsilon_k + \varsigma_k^{\top}\Psi\varsigma_k, \right) \qquad (26)$$

over $W_{\mathrm{x}}, W_{\mathrm{y}}$, where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm. Matrices $\Gamma = \mathrm{diag}(\gamma_1, \ldots, \gamma_n)$ and $\Psi = \mathrm{diag}(\psi_1, \ldots, \psi_{n_y})$ are diagonal positive weighting matrices on these residuals, and are known as regularization matrices. This problem can be solved in the dual form by introducing the *Lagrangian* as

$$\mathcal{K}(W_{\mathrm{x}}, W_{\mathrm{y}}, \alpha, \beta, \varepsilon, \varsigma) = \mathcal{I} - \sum_{j=1}^N \beta_j^{\top}\left(W_{\mathrm{y}}\varphi_{\mathrm{y}}^{\top}(p_j) + \varsigma_j - y_j \right)$$
$$- \sum_{j=1}^N \alpha_j^{\top}\left(W_{\mathrm{x}}\varphi_{\mathrm{x}}^{\top}(p_j) + \varepsilon_j - \check{x}_{j+1} \right), \qquad (27)$$

where $\alpha_j \in \mathbb{R}^n$, $\beta_j \in \mathbb{R}^{n_y}$ are the Lagrange multipliers at time j. The solution is obtained by solving the KKT conditions, i.e., equating the partial derivatives $\frac{\partial \mathcal{K}}{\partial W_{\mathrm{x}}}, \frac{\partial \mathcal{K}}{\partial W_{\mathrm{y}}}, \frac{\partial \mathcal{K}}{\partial \alpha_j}, \frac{\partial \mathcal{K}}{\partial \beta_j}, \frac{\partial \mathcal{K}}{\partial \varepsilon_j}, \frac{\partial \mathcal{K}}{\partial \varsigma_j}$ to zero. For brevity, we do not list the KKT conditions here (see Rizvi et al., 2015a). Substituting these conditions in (25a)–(25b), we can eliminate the

---

[4] Compared to the possible case of dependencies in (21), for the sake of simplicity, here we restrict our dependency class to be static (as in the original representation form (2)).

primal decision variables and obtain the following equations:

$$\check{x}_{k+1} = \underbrace{\left\{ \sum_{j=1}^{N} \alpha_j \varphi_x(p_j) \right\}}_{W_x} \varphi_x^\top(p_k) + \underbrace{\Gamma^{-1} \alpha_k}_{\varepsilon_k}, \tag{28a}$$

$$y_k = \underbrace{\left\{ \sum_{j=1}^{N} \beta_j \varphi_y(p_j) \right\}}_{W_y} \varphi_y^\top(p_k) + \underbrace{\Psi^{-1} \beta_k}_{\varsigma_k}. \tag{28b}$$

By replacing the inner-product $\Phi_i^\top(p_k)\Phi_i(p_j)$ by a kernel function $\bar{k}_i(p_j, p_k)$ and defining

$$[\Omega]_{j,k} = \sum_{i=1}^{3} z_i^\top(j) \bar{k}_i(p_j, p_k) z_i(k), \tag{29a}$$

$$[\Xi]_{j,k} = \sum_{i=4}^{5} z_i^\top(j) \bar{k}_i(p_j, p_k) z_i(k), \tag{29b}$$

where $\quad z_i(k) = \begin{cases} \check{x}_k, & i = 1, 4 \\ u_k, & i = 2, 5 \\ y_k, & i = 3. \end{cases}$

We can now write (28) in a compact form as follows:

$$\check{X}_{k+1} = \alpha \Omega + \Gamma^{-1} \alpha, \tag{30a}$$

$$Y = \beta \Xi + \Psi^{-1} \beta, \tag{30b}$$

where $\Omega \in \mathbb{R}^{N \times N}$ and $\Xi \in \mathbb{R}^{N \times N}$ are kernel matrices as defined above, $\alpha = [\alpha_1 \cdots \alpha_N] \in \mathbb{R}^{n \times N}$ and $\beta = [\beta_1 \cdots \beta_N] \in \mathbb{R}^{n_y \times N}$ are the matrices containing the Lagrange multipliers, $\check{X}_{k+1} = [\check{x}_2 \cdots \check{x}_{N+1}] \in \mathbb{R}^{n \times N}$ and $Y = [y_1 \cdots y_N] \in \mathbb{R}^{n_y \times N}$ contain the estimated states and outputs for the N samples, respectively. The solution to the above equations can be obtained as follows:

$$\text{vec}(\alpha) = \left(I_N \otimes \Gamma^{-1} + \Omega^\top \otimes I_n \right)^{-1} \text{vec}(\check{X}_{k+1}), \tag{31a}$$

$$\text{vec}(\beta) = \left(I_N \otimes \Psi^{-1} + \Xi^\top \otimes I_{n_y} \right)^{-1} \text{vec}(Y), \tag{31b}$$

where matrices I_N and I_{n_y} represent identity matrices of dimensions N and n_y, respectively. For a given solution to (31a)–(31b), the estimate of the state-space matrices can be calculated by using the KKT conditions in (25a)–(25b) as

$$\tilde{A}_e(\cdot) = W_1 \Phi_1(\cdot) = \sum_{j=1}^{N} \alpha_j \check{x}_j^\top \bar{k}_1(p_j, \cdot), \tag{32a}$$

$$\tilde{B}_e(\cdot) = W_2 \Phi_2(\cdot) = \sum_{j=1}^{N} \alpha_j u_j^\top \bar{k}_2(p_j, \cdot), \tag{32b}$$

$$C_e(\cdot) = W_4 \Phi_4(\cdot) = \sum_{j=1}^{N} \beta_j \check{x}_j^\top \bar{k}_4(p_j, \cdot), \tag{32c}$$

$$K_e(\cdot) = W_3 \Phi_3(\cdot) = \sum_{j=1}^{N} \alpha_j y_j^\top \bar{k}_3(p_j, \cdot), \tag{32d}$$

$$D_e(\cdot) = W_5 \Phi_5(\cdot) = \sum_{j=1}^{N} \beta_j u_j^\top \bar{k}_5(p_j, \cdot). \tag{32e}$$

This gives a nonparametric estimate of the SS matrices.

5. Tuning of the hyper-parameters

Both the state trajectory estimation in terms of the KCCA problem and the estimation of the matrix functions of the state-space representation require the choice of hyper-parameters for the definition of the associated kernel functions and other regularization parameters. Let θ_c be the collection of hyper-parameters: ν_f, ν_p and kernel coefficients, e.g., σ_c associated with the state estimation detailed in Section 3, while θ_s be the collection of hyper-parameters: $\gamma_1, \ldots, \gamma_n, \psi_1, \ldots, \psi_{n_y}$ and kernel coefficients, e.g., σ_s, associated with the matrix function estimation problem in Section 4. Let $\theta = [\theta_c^\top \; \theta_s^\top]^\top$.

Denote the model $\mathcal{M}(\theta)$ as the solution of the specified state and matrix function estimation problems using the estimation data set $\mathcal{D}$ and a fixed choice of θ. Additionally, let $\mathcal{D}_{\text{val}}$ be an independent data set generated by (1) and define

$$\text{BFR}(\theta) = 100\% \cdot \max \left(1 - \frac{\| y_k - \hat{y}_k(\theta) \|_2}{\| y_k - \bar{y} \|_2}, 0 \right), \tag{33}$$

as fitness score or *best fit rate* (BFR) between the actual output trajectory y of $\mathcal{D}_{\text{val}}$, its mean $\bar{y}$, and the simulated[5] output $\hat{y}$ of the identified model $\mathcal{M}(\theta)$ *w.r.t.* the inputs and scheduling trajectory of $\mathcal{D}_{\text{val}}$. We seek to maximize (33) over θ. This results in a nonlinear optimization problem with a quadratic cost function, which can be seen as an inference problem between the data sets $\mathcal{D}$ and $\mathcal{D}_{\text{val}}$ under the given parametrization of the estimation problems in terms of θ.

Alternatively, we also formulate the choice of the hyper-parameters θ_c and θ_s under a Bayesian setting by assuming a Gaussian distribution (with an RBF kernel) of the state variation and matrix functions and a uniform distribution of θ_c and θ_s. Note that in this setting, the matrix function estimation problem (Section 4) is conditioned on the estimated $\check{x}$, dependent on the prior θ_c, and as the estimation involves an SVD, even in case of a fixed choice of n no joint formulation of the hyper-parameter estimation of θ_c and θ_s can be given. Furthermore, Hardoon and Shawe-Taylor (2009) provides a detailed theoretical analysis of KCCA from the stochastic point of view. However, the problem of choosing the regularization parameter even under a Gaussian setting remains largely unsolved without using any approximation of the resulting expressions. These problems somewhat undermine the stochastic efficiency of the resulting methodology. Nevertheless, for illustration purposes the choice of θ_s is formulated in terms of a log *marginal likelihood* (ML) function conditioned on it. Following the work in Rasmussen and Williams (2006), and using (30), we define the log marginal likelihood function as

$$\log \bar{p}(Y|\check{X}, U, P, \theta_s) = -\frac{1}{2} \left(\sum_{i=1}^{n_y} Y_i \Xi_i^{-1} Y_i^\top + \log|\Xi_i| \right)$$
$$- \frac{1}{2} \left(\sum_{i=1}^{n_x} \check{X}_i \Omega_i^{-1} \check{X}_i^\top + \log|\Omega_i| \right) - \frac{N}{2} \log 2\pi, \tag{34}$$

where the kernel matrices Ξ_i and Ω_i are defined as $\Xi_i = \Xi + \psi_i^{-1} I_N$, $\Omega_i = \Omega + \gamma_i^{-1} I_N$ and $Y_i, \check{X}_i$ denote the ith output/state or ith row of Y and $\check{X}$, $\bar{p}$ denotes probability, and Y, U, P, and $\check{X}$ represent the measurements of outputs, inputs, scheduling variables, and the estimated states, respectively. By maximizing the log ML function (34) over the parameter set θ_s, we seek the set of hyper-parameters that maximize the likelihood of fitting the estimated LPV-SS model outputs to the given observations Y and $\check{X}_i$, corresponding to a nonlinear optimization problem over $\check{\mathcal{D}}$.

---

[5] Alternatively, one can formulate a similar objective function *w.r.t.* the predicted output of $\mathcal{M}(\theta)$ using the predictor form (2).

6. Numerical examples

Example 1. Consider the following discrete-time data-generating system

$$x_{k+1} = A(p_k)x_k + B(p_k)u_k + K(p_k)e_k,$$
$$y_k = C(p_k)x_k + e_k$$

with

$$A(p_k) = \begin{bmatrix} \text{sat}(p_k) & 1 & 0 & 0 \\ \dfrac{1}{2} & \dfrac{p_k^3}{8} & \dfrac{4}{10} & \dfrac{1}{5} \\ \dfrac{3}{10} & 0 & \dfrac{p_k^2}{5} & \dfrac{1}{8} \\ 0 & 0 & \dfrac{1}{2} & \dfrac{1}{5} \end{bmatrix},$$

$$K(p_k) = \begin{bmatrix} \dfrac{\tanh(p_k)}{3p_k} & 0 \\ 0 & 0 \\ 0 & \sin(2\pi p_k) + \cos(2\pi p_k) \\ 0 & 1 \end{bmatrix},$$

$$C(p_k) = \begin{bmatrix} \dfrac{p_k^2}{5} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad B(p_k) = \begin{bmatrix} \dfrac{p_k^4}{5} & 0 & \dfrac{1}{5} & 0 \end{bmatrix}^\top,$$

where $\text{sat}(p_k)$ is a saturation function with limits at ± 0.5 and unity slope. Given the measurements u_k, y_k, and p_k in $\mathcal{D}$, we want to estimate the matrix functions $A, \ldots, K$. The data-generating system with initial condition $x_0 = [0\ 0\ 0\ 0]^\top$ has been simulated with uniformly distributed input signal $u_k \sim \mathcal{U}(-1, 1)$, $p_k = \sin(0.3k)$ and normally distributed measurement noise $e_k \sim \mathcal{N}(0, I\sigma_e^2)$ to generate a data set $\mathcal{D} = \{u_k, y_k, p_k\}_{k=1}^N$ with $N = 1100$, where σ_e^2 has been chosen to guarantee a 20 dB *signal-to-noise ratio* (SNR). The data is divided into 800 and 300 samples for estimation $\mathcal{D}$ and validation $\mathcal{D}_{\text{val}}$, respectively. Polynomial kernel is chosen for the state estimation step and the past and future window size of $d = 4$ is selected while for the matrix function estimation problem RBF kernels are employed. Other kernels have also been tested and the selection of the kernel, and its associated hyper-parameters, is made based on the minimization of the cross-validation objective function (33). This optimization problem is solved[6] using the `fmincon` solver in MATLAB. Followed by this, the ML approach corresponding to the maximization of (34) is also implemented. While the solution θ depends on initialization of the optimization problem, we observe that in the best case, the maximization of likelihood function (34) gives us comparable, and in some cases, slightly improved solution. The proposed kernel CCA-based algorithm is run and the state sequence is estimated. The order of the system is selected by solving the SVD problem (20). A plot of the first 50 singular values $\tilde{\sigma}_j$, $j = 1, \ldots, 50$, is shown in Fig. 1. We observe a significant gap between the first four singular values and the next four that follow them. Using the extended data set $\check{\mathcal{D}} = \{u_k, y_k, \check{x}_k, p_k\}_{k=1}^N$, we then run the LS-SVM identification algorithm of Section 4 based on different choices of system order to estimate the state-space matrices. An RBF kernel is chosen to find an estimate of the matrix functions. For independently generated data sets, the estimation is repeated 100 times in a Monte-Carlo study and the fitness score statistics are tabulated in Table 1.
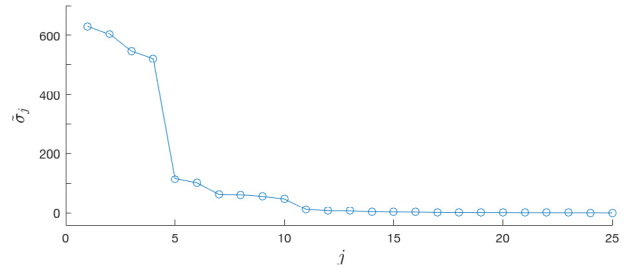
[6] For $\hat{n} = 8$, the optimization problem resulted in $\nu_f = 1000$, $\nu_p = 8200$, deg $= 2$ for the polynomial kernel, $\{\sigma_{s,i}\}_{i=1}^4 = \{1.05, 10.15, 2 \times 10^{-4}, 9.1\}$ for the RBF kernel, $\{\gamma_i\}_{i=1}^8 = \{700, 400, 750, 700, 500, 1400, 1400, 720\}$ and $\{\psi_i\}_{i=1}^2 = \{5500, 0.5\}$.

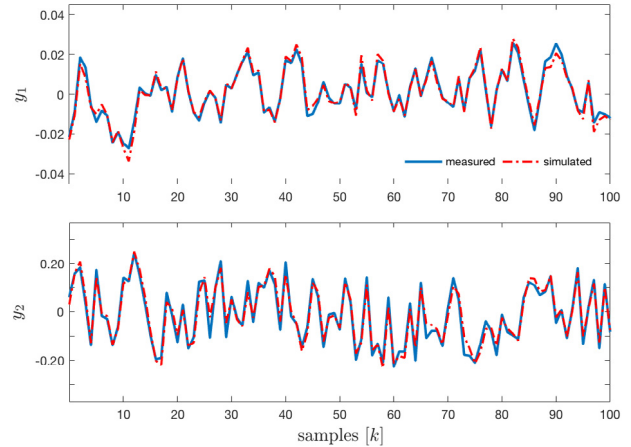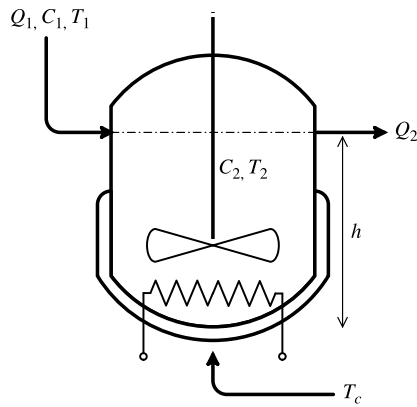**Fig. 1.** Singular values of the SVD problem (20).

Fig. 2. Simulated output response of the estimated LPV model in the given example computed on $\mathcal{D}_{\text{val}}$ (red) and the noise free response of the original system (blue); for the sake of clarity, only 100 of the validation data points are shown here. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Monte-Carlo simulation results for Example 1.

	$\hat{n}$	Mean (BFR %)	Std. (BFR %)
SNR 25 dB	4	85.15	1.12
	8	87.03	0.751
SNR 20 dB	4	83.91	0.911
	8	86.31	0.022
	9	86.03	1.015

We observe that by selecting the order to be $\hat{n} = 8$, a slightly higher fitness rate is obtained compared to $\hat{n} = 4$. Note that this can be explained by the fact that the non-smooth saturation function based state-map is difficult to be captured by a polynomial kernel. For $\hat{n} = 9$ and onwards, no significant improvement is observed in terms of accuracy of the simulated outputs. This is also corroborated by the singular values plot shown in Fig. 1. Fig. 2 shows the simulated outputs of an identified model (dashed red line) in the Monte-Carlo run compared to the original noise-free outputs of the data-generating system (solid blue line). In order to quantify how well the simulated outputs of the estimated LPV-SS model fit the actual outputs, the BFR is evaluated on the validation data set $\mathcal{D}_{\text{val}}$ and the statistics are tabulated in Table 1. Obtained average BFR values with small standard deviation demonstrate consistent performance of the proposed LPV model identification algorithm.

Fig. 3. An ideal continuous stirred tank reactor.

Fig. 4. Scheduling trajectory $C_1(\text{kg/m}^3)$ for Example 2.

Table 2
CSTR output fitness simulation results.

	SNR (dB)	BFR (%)
LS-SVM (full states measurement)	25	86.72
KCCA-based LS-SVM	25	83.23

Example 2. The model of an ideal *continuous stirred tank reactor* (CSTR) is considered here. Schematic diagram of the CSTR process is shown in Fig. 3. It shows the chemical reaction that converts an inflowing liquid to a product; this reaction is non-isothermal as described in Tóth, Van den Hof, Ludlage, and Heuberger (2010). The first principles-based model is described by

$$\dot{T}_2 = \frac{Q_1}{V}(T_1 - T_2) - \frac{U_{\text{HE}}}{A_{\text{HE}}}(T_2 - T_c) + \frac{\Delta H k_0}{\rho c_\rho} e^{-\frac{E_A}{R T_2}} C_2,$$

$$\dot{C}_2 = \frac{Q_1}{V}(C_1 - C_2) - k_0 e^{-\frac{E_A}{R T_2}} C_2, \tag{35}$$

where T_c, T_1, T_2 are the temperatures for the coolant, the inflowing and outflowing liquids, and C_1, C_2 denote the concentration of the raw material and the product, respectively. Variable Q_1 denotes the input flow of raw material to the CSTR. A typical control objective is often to regulate the concentration C_2 or the temperature T_2 in the reactor; these two signals make up the internal states of the system. In this study, we consider the product temperature T_2 to be the regulated output. Variables Q_1 and T_c are used as manipulatable signals. Steady-state operating conditions and parameter values are taken from Tóth et al. (2010) and are not reproduced here. Raw material concentration C_1 is taken as the scheduling variable p. This is because step changes in the manipulated variables at different values of C_1 show significantly different dynamics in terms of time constants, relative gains, and even the sign of the gain exhibits non-minimum-phase behavior at certain operating points (Tóth et al., 2010). The process is assumed to be a first order reaction with a temperature relation according to Arrhenius law; it is also assumed that the temperature increase in the coolant over the coil can be neglected. The inflow and outflow rates, Q_1 and Q_2, are kept equal. Previously, we published our LPV-SS identification study of the CSTR model in Rizvi et al. (2015b) under the strong assumption that both the internal states C_2, T_2 were available for full measurement. Here, we intend to assess the capabilities of our KCCA-based LS-SVM algorithm to predict the internal states when they are not available in full for measurements, and then use them for LPV identification. We then intend to compare the performance with our previous results that assumed full measurements of states. For this purpose, the measurements for C_2 are assumed not available, and only output measurements T_2 are taken. *Pseudo random binary sequences* (PRBS) of the two inputs with $\pm$ 10% of the nominal values are used to excite the CSTR model. Gaussian white noise is added such that 25 dB SNR is maintained for the output T_2. A slowly-varying trajectory for the scheduling variable C_1 with limits at $\pm 50\%$ of the nominal value is used, as shown in Fig. 4. A
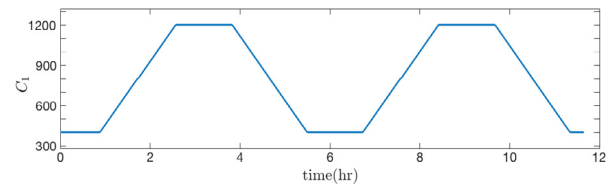
sampling time of 60 s is chosen to generate the input–output data. We choose RBF kernel for the KCCA-based state estimation, and make use of optimization of (34) to obtain the hyper-parameter set.[7] The past and future window size for the input, output and scheduling variables data is set to $d = 4$. The KCCA-based LS-SVM algorithm estimates state sequences $\check{x}$ and then uses the augmented data set to find the state-space matrix functions. The fitness scores based on BFR percentage (33) are also calculated, and their values are presented in Table 2. Compared to the LS-SVM algorithm that assumes complete measurement of both states, the KCCA-based LS-SVM algorithm obtains slightly lower, but comparable BFR values. However, the proposed kernelized CCA algorithm does well to find directions in the RKHS where correlation between inputs and outputs is maximized, and then uses this solution to estimate transformed estimates of the states. This gives us, despite the lack of state measurements in full, a comparable prediction for T_2. Negligibly small improvements are observed if an attempt is made to fit a higher order model. Simulated values of the identified models with and without state measurements are shown in Fig. 5. Overall, the proposed kernelized CCA for state estimation in LPV-SS models shows an encouraging ability to estimate state variables that can tie past and future input–output behavior together.

7. Concluding remarks

This paper has presented a nonparametric method for identification of LPV-SS models. The proposed technique relies only on the inputs, outputs, and scheduling variables data. The states are estimated up to a similarity transformation by using correlation analysis between the past and future data. Once estimated, an LS-SVM-based non-parametric scheme is used to identify the underlying LPV model. The proposed scheme solves a convex optimization problem and provides encouraging results on a MIMO numerical example with challenging nonlinearities in the presence of noise. The proposed algorithm is further validated on the model of a continuous stirred tank reactor process, and results are compared with an earlier study that assumes complete knowledge of the states. We find that kernel CCA provides encouraging state reconstruction results, which can then be augmented with the measured data in order to build an LPV-SS model. The main contribution of this paper lies in formulating the kernel CCA and LS-SVM solution for

---

7 For $\hat{n} = 2$, the tuned values of the hyper-parameters are as follows: $\sigma_c = 470$, $\nu_f = 1000$, $\nu_p = 1000$, $\{\sigma_{s,i}\}_{i=1}^4 = \{360, 2600, 360, 7000\}$, $\{\gamma_i\}_{i=1}^2 = \{500, 500\}$ and $\psi = 1.2 \times 10^5$.

Fig. 5. Example 2: (left) Validation results for CSTR output temperature $T_2(^\circ C)$ using LS-SVM-based identification with and without full states measurements, and (right) validation error.

this identification problem by preserving the linearity structure in parameter-dependent state-space models. The proposed method also does not impose any dependency structure on the matrix functions, affine or otherwise. Since LPV-SS models are important for LPV control synthesis purposes, we believe that this work has the potential to pave the way for efficient low-order LPV modeling for control synthesis.

References

Abbasi, F., Mohammadpour, J., Tóth, R., & Meskin, N. (2014). A support vector machine-based method for LPV-ARX identification with noisy scheduling parameters. In *Proc. of the 13th European control conf.* (pp. 370–375).

Bach, F. R., & Jordan, M. I. (2003). Kernel independent component analysis. *Journal of Machine Learning Research (JMLR)*, 3, 1–48.

Bamieh, B., & Giarre, L. (2002). Identification of linear parameter varying models. *International Journal of Robust and Nonlinear Control*, 12(9), 841–853.

Cucker, F., & Smale, S. (2001). On the mathematical foundations of learning. *American Mathematical Society. Bulletin*, 39(1), 1–49.

De Bie, T., & De Moor, B. (2003). On the regularization of canonical correlation analysis. In *Int. symp. independent component analysis and blind signal separation* (pp. 785–790).

dos Santos, P. L., Azevedo-Perdicoulis, T. P., Ramos, J. A., Deshpande, S., Rivera, D. E., & de Carvalho, J. L. M. (2014). LPV system identification using a separable least squares support vector machines approach. In *Proc. of the 53rd IEEE conf. decision and control* (pp. 2548–2554).

Felici, F., van Wingerden, J. W., & Verhaegen, M. (2007). Subspace identification of MIMO LPV systems using a periodic scheduling sequence. *Automatica*, 43(10), 1684–1697.

Hardoon, D. R., & Shawe-Taylor, J. (2009). Convergence analysis of kernel canonical correlation analysis: theory and practice. *Machine Learning*, 74(1), 23–38.

Laurain, V., Tóth, R., Zheng, W. X., & Gilson, M. (2012). Nonparametric identification of LPV models under general noise conditions: an LS-SVM based approach. In *Proc. of the 16th IFAC symposium on syst. identification* (pp. 1761–1766).

Overschee, P. V., & DeMoor, B. (1995). Choice of state-space basis in combined deterministic-stochastic subspace identification. *Automatica*, 31(12), 1877–1883.

Rasmussen, C. E., & Williams, C. K. (2006). Model selection and adaptation of hyperparameters. In *Gaussian processes for machine learning. 2006* (pp. 105–128). Cambridge, MA: The MIT Press, (Chapter 5).

Rizvi, S. Z., Mohammadpour, J., Tóth, R., & Meskin, N. (2015a). An IV-SVM-based approach for identification of state-space LPV models under generic noise conditions. In *Proc. of the 54th IEEE conf. decision and control* (pp. 7380–7385). Osaka, Japan.

Rizvi, S. Z., Mohammadpour, J., Tóth, R., & Meskin, N. (2015b). A Kernel-based approach to MIMO LPV state-space identification and application to a nonlinear process system. In *Proc. of the 1st IFAC workshop on linear parameter varying systems* (pp. 85–90). Grenoble, France.

Schölkopf, B., & Smola, A. (2002). *Learning with Kernels*. Cambridge, MA: MIT Press.

Suykens, J. A. K, Gestel, T. V., Moor, B. De, & Vandewalle, J. (2002). Basic methods of least squares support vector machines. In *Least squares support vector machines*. World Scientific.

Tóth, R. (2010). *Modeling and identification of linear parameter-varying systems*. Heidelberg: Springer.

Tóth, R., Abbas, H. S., & Werner, H. (2012). On the state-space realization of LPV input-output models: Practical approaches. *IEEE Transactions on Control Systems Technology*, 20(1), 139–153.

Tóth, R., Laurain, V., Zheng, W. X., & Poolla, K. (2011). Model structure learning: A support vector machine approach for LPV linear-regression models. In *Proc. of the 50th IEEE conf. decision and control, and european control conf.* (pp. 3192–3197). Dec.

Tóth, R., Van den Hof, P. M. J., Ludlage, J. H. A., & Heuberger, P. S. C. (2010). Identification of nonlinear process models in an LPV framework. In *Proc. of the 9th Int. symp. dynamics and control of process syst.*, Leuven, Belgium.

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

Verdult, V., Suykens, J. A. K., Boets, J., Goethals, I., & De Moor, B. (2004). Least squares support vector machines for kernel CCA in nonlinear state-space identification. In *Proc. of the 16th Intl. symp. math. theory of networks and syst*, Leuven, Belgium.

Verdult, V., & Verhaegen, M. (2002). Subspace identification of multivariable linear parameter-varying systems. *Automatica*, 38(5), 805–814.

Verdult, V., & Verhaegen, M. (2005). Kernel methods for subspace identification of multivariable LPV and bilinear systems. *Automatica*, 41(9), 1557–1565.

Syed Zeeshan Rizvi obtained his B.E. in Electronics Engineering from N.E.D. University of Engineering & Tech., Pakistan and his M.S. in Electrical Engineering from King Fahd University of Petroleum & Minerals, Saudi Arabia, in years 2006 and 2008, respectively. In January 2013, he joined the Complex Systems Control Laboratory at The University of Georgia in Athens, GA, where he focused on system identification, model reduction, and control synthesis methods for linear parameter-varying models. He obtained his Ph.D. focusing on LPV system identification and control in December 2016. He is currently working as a process control scientist with Corning Inc. in Corning, NY, focusing on data analytics and advanced process control solutions for specialty glass and ceramic manufacturing processes.

Javad Mohammadpour Velni received B.S. and M.S. degrees in electrical engineering from Sharif University of Technology and University of Tehran, Iran, respectively, and Ph.D. degree in mechanical engineering from University of Houston, TX. He joined the University of Georgia as an assistant professor of electrical engineering in August 2012. Prior to that, he was with the University of Michigan, where he worked in the naval architecture & marine engineering department from October 2011 to July 2012. He was also a Research Assistant Professor of mechanical engineering at University of Houston from October 2008 to September 2011 and a Research Associate at the same institution from January 2008 to September 2008. He has published over 100 articles in international journals and conference proceedings, served in the editorial boards of ASME and IEEE conferences on control systems and edited two books on control of large-scale systems (published in 2010) and LPV systems modeling, control and applications (published in 2012). His current research interests are in secure control of cyber physical systems (and in particular, smart grids), coverage control of heterogeneous multi-agent systems, and data-driven approaches for model learning and control of complex distributed systems.

Farshid Abbasi received his B.Sc. and M.Sc. degrees in Mechanical Engineering both from University of Tabriz in 2007 and 2010, respectively. In January 2013, he joined Complex Systems Controls Lab at the University of Georgia, where he received a Ph.D. in dynamic system and controls in December 2016. His research interests include multi-agent systems, cooperative control, machine learning and system identification methods focusing on complex nonlinear processes. He is currently with ASML as a controls and system identification research scientist where his research focuses on developing new control and data analysis techniques to enhance rapidly changing semiconductor technologies.

Roland Tóth was born in 1979 in Miskolc, Hungary. He received the B.Sc. degree in Electrical Engineering and the M.Sc. degree in Information Technology in parallel with distinction at the University of Pannonia, Veszprém, Hungary, in 2004, and the Ph.D. degree (cum laude) from the Delft Center for Systems and Control (DCSC), Delft University of Technology (TUDelft), Delft, The Netherlands, in 2008. He was a Post-Doctoral Research Fellow at DCSC, TUDelft, in 2009 and at the Berkeley Center for Control and Identification, University of California Berkeley, in 2010. He held a position at DCSC, TUDelft, in 2011–2012. Currently, he is an Assistant Professor at the Control Systems Group, Eindhoven University of Technology (TU/e). He is an Associate Editor of the IEEE Conference Editorial Board, the IEEE Transactions on Control Systems Technology and the International Journal of Robust and Nonlinear Control. His research interests are in linear parameter-varying (LPV) and nonlinear system identification, machine learning, process modeling and control, model predictive control and behavioral system theory. He received the TUDelft Young Researcher Fellowship Award in 2010, the VENI award of The Netherlands Organisation for Scientific Research in 2011 and the Starting Grant of the European Research Council in 2016.

Nader Meskin received his B.Sc. from Sharif University of Technology, Tehran, Iran, in 1998, his M.Sc. from the University of Tehran, Iran in 2001, and obtained his Ph.D. in Electrical and Computer Engineering in 2008 from Concordia University, Montreal, Canada. He was a postdoctoral fellow at Texas A&M University at Qatar from January 2010 to December 2010. He is currently an Associate Professor at Qatar University and Adjunct Associate Professor at Concordia University, Montreal, Canada. His research interests include Fault Detection and Isolation (FDI), multi-agent systems, active control for clinical pharmacology, and linear parameter varying systems. He has published more than one hundred refereed journal and conference papers and he is a coauthor (with K. Khorasani) of the book Fault Detection and Isolation: Multi-Vehicle Unmanned Systems (Springer 2011).