

Affine Linear Parameter-Varying Embedding of Nonlinear Models with Improved Accuracy and Minimal Overbounding

 ISSN 1751-8644
 doi: 0000000000
 www.ietdl.org

 Arash Sadeghzadeh¹, Bardia Sharif², Roland Tóth^{1,3}
¹ Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

² Control Systems Technology Group, Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

³ Systems and Control Laboratory, Institute for Computer Science and Control, Budapest, Hungary

Abstract: In this paper, automated generation of linear parameter-varying (LPV) state-space models to embed the dynamical behavior of nonlinear systems is considered, focusing on the trade-off between scheduling complexity and model accuracy and on the minimization of the conservativeness of the resulting embedding. The LPV state-space model is synthesized with affine scheduling dependency, while the scheduling variables themselves are nonlinear functions of the state and input variables of the original system. The method allows to generate complete or approximative embedding of the nonlinear system model and also it can be used to minimize complexity of existing LPV embeddings. The capabilities of the method are demonstrated on simulation examples and also in an empirical case study where the first-principle motion model of a 3-DOF control moment gyroscope is converted by the proposed method to an LPV model with low scheduling complexity. Using the resulting model, a gain-scheduled controller is designed and applied on the gyroscope, demonstrating the efficiency of the developed approach.

1 Introduction

The *linear parameter-varying* (LPV) framework has been introduced to tackle the control problem of *nonlinear* (NL) and *time-varying* (TV) systems using the extension of powerful linear control methods [1]. This methodology offers great potential in a wide variety of practical applications [2]. By using so-called *scheduling variables*, which express nonlinear or time-varying behavior, LPV systems are capable of representing the solution set of NL/TV systems in terms of a linear structure. Extension of *linear-time invariant* (LTI) methods to exploit this linear proxy representation has seen a tremendous development [2]; however, the lack of systematic methods that are capable of automatically deriving LPV embeddings of NL/TV systems has prevented the widespread industrial use of the LPV concept [3].

While LPV system identification methods have matured over the last 15 years with many competitive approaches, e.g. [3–10] to mention a few, conversion methods of existing NL/TV models of applications has seen only moderate progress. As in practice, often high-fidelity models of the target application are available due to the development and design process, e.g. rigid body/flexible motion dynamics in mechatronic applications. The need for embedding approaches in which such models are converted to LPV description to be used for control design or prediction purposes is of great significance.

Existing methods can be categorized as local and global methods. In local LPV model conversion, an NL description of the system is linearized at several operating points, then the obtained linearized models are interpolated to get an LPV model. However, due to the local information on the dynamic aspects, closed-loop stability and performance cannot be guaranteed by controllers, designed based on the resulting LPV models, unless the variation of the scheduling variable is guaranteed to be sufficiently slow, see [3, 9] for an overview. In global methods, the NL/TV system model is directly converted to an LPV representation such that the original system behavior is embedded in the solution set of the resulting LPV model. The methods can be categorized as *substitution-based transformation methods* and (automated) *model transformation methods*. In substitution based transformation techniques, NL terms are considered to be absorbed by the introduced scheduling

variables, which results in a global LPV embedding of NL dynamics; nevertheless, the applicability of these approaches is either limited to a narrow class of NL systems (e.g., [11, 12]) or the methods are based on commonly used, but ad-hoc substitutions (e.g., [13, 14]). In the so-called velocity linearization [15], differentiating the NL state-space model of a system, a representation in terms of derivatives of the inputs, outputs, and state variables multiplied by some nonlinear functions are obtained. Considering these nonlinearities as the scheduling, an LPV model is developed, but requiring specialized control synthesis methods. Model transformation methods are based on the systematic exploration of possible ways of reformulating the NL system as an LPV model with the smallest possible conservatism. Next to the computationally intensive methods in [3], recent developments include approaches based on *linear fractional representation* (LFR) with a nonlinear feedback block converted to an LPV model depending affinely on the scheduling variables in [16]. Choosing a LPV embedding for nonlinear systems is investigated in [17] by minimizing the projection of the nonlinearities onto directions deleterious for performance. The later problem is cast as a computationally intensive *linear matrix inequality* (LMI) based optimization. A systematic embedding method to achieve a state-minimal LPV representation in the observable canonical form is presented in [18]. Besides the problem of LPV embedding, reduction of complexity in sense of reducing the number of scheduling variables, simplification of the dependency on the scheduling variables, and tightening the admissible region of the scheduling variables have received attention in recent years. In [19], taking advantage of principal component analysis (PCA) applied on the typical scheduling trajectories, a method is proposed to obtain LPV models with fewer scheduling variables. It is alleged that the procedure can lead to less overbounding of admissible regions for the scheduling variables without providing a rigorous proof. As an extension to that method, a *linear fractional transformation* (LFT)-based LPV representation for descriptor systems is proposed in [20]. The drawback related to these approaches is that they mainly focus on the scheduling variables not their effects on the dynamical behavior of the system. An approach based on Ho-Kalman algorithm is presented in [21] to address the problem of joint state-order and scheduling-dependency reduction, but applicability of this approach is limited to small scale problems.

In summary, next to accurate representation of the behavior of the NL/TV system, the three main challenges that LPV embedding methods need to face with are (i) *scheduling complexity* minimization, (ii) minimization of *conservativeness* of the embedding, (iii) preservation of *structural properties* like controllability, stability, etc. Most LPV controller synthesis methods rely on linear or quadratic optimization with LMI constraints [22–28], the number of which grows exponentially with the scheduling dimension/number of vertices used to describe the scheduling range. Therefore, in terms of problem (i), achieving a minimum number of scheduling variables in LPV modeling has paramount importance in practice. Moreover, many methods are formulated for LPV state-space representations with *affine dependence* on the scheduling variables, while for mechatronic and chemical systems, straightforward manual conversion results in rational or even exponential dependence that is often hidden in new scheduling variables, leading to at least doubling of the scheduling dimension. Regarding problem (ii), the scheduling variables are usually assumed to vary independently in some specified ranges; while as the above discussion exemplify it, they are functions of some measurable variables of the system with often complicated nonlinear dependence. Thus, the scheduling variable dependency in practice leads to conservativeness of LPV models due to the fact that represented solutions include the solution set of the embedded nonlinear system plus those trajectories that result due to forgetting the above mentioned dependence. This is the price to be paid for a linear representation of the dynamics, but excessive conservativeness can lead to degradation of the achievable performance or even feasibility by LPV control as all extra dynamics resulting purely from conversion are needed to be stabilized and shaped during control synthesis. To the best of our knowledge, a general approach for LPV embedding of nonlinear systems in which the aforementioned factors (i)-(iii) are all appropriately addressed is not available yet. This paper aims to address problems (i)-(ii) and balance complexity, conservativeness and accuracy in LPV model conversion in terms of a practically applicable method.

Three typical problem settings connected to model conversion are considered, namely (a) embedding of nonlinear systems to obtain LPV state-space representation with affine dependence on the scheduling variables and minimal conservativeness; (b) simplifying the dependency of an LPV model depending nonlinearly on the scheduling variables to an affine LPV state-space representation with minimal conservativeness; (c) constructing an affine LPV model with restricted number of scheduling variables from an affine LPV model having too many scheduling variables. Our first contribution is to show that these problem formulations can be uniformly expressed as a single realization problem. Then, by deriving an extension of the approach in [19], we apply *principle component analysis* (PCA) to solve the minimal scheduling variable realization problem under affine dependency of the resulting LPV state-space form using a bundle of generated state and input trajectories of the system along which the variation of state-equations are expressed. Contrary to the method of [19] in which the PCA is applied on the data matrix consisting of the individual scheduling variable trajectories, our contribution is to consider variation of the state-equations directly which, as shown through examples, leads to further reduction of the scheduling complexity. Additionally, an accuracy index is defined to address the trade-off between the number of scheduling variables and the model accuracy, which facilitates determining the number of scheduling variables that are required for the embedding. To minimize conservativeness, we optimize the scheduling range in terms of a minimal hyper-rectangle. Our contribution is to formulate this scheduling range minimization problem and the connected rotational problem of the scheduling space. To reveal the advantages of the proposed method over the existing approaches, the presented method in this paper is applied on simulation examples and validated empirically on a 3-DOF gyroscope system. In the later case, using the obtained LPV model, a full-order gain-scheduled output feedback controller is designed on the converted low complexity model and verified on the experimental setup.

Notation: $L_{i,j} \in \mathbb{R}$ denotes the elements of matrix $L \in \mathbb{R}^{m \times n}$, i.e. $L := [L_{i,j}]_{m \times n}$. $\vec{\Gamma}$ refers to row-wise vectorization of L :

$$\vec{L} := [L_{1,1} \cdots L_{1,n} \quad L_{2,1} \cdots L_{2,n} \quad \cdots \quad L_{m,1} \cdots L_{m,n}]^T.$$

For $X = \vec{L} \in \mathbb{R}^{(mn) \times 1}$, the reverse operation is

$$L = \underset{m \times n}{\leftarrow} X \in \mathbb{R}^{m \times n}.$$

The notation $\|L\|_F := \sqrt{\sum_{i=1}^m \sum_{j=1}^n |L_{i,j}|^2}$ corresponds to the Frobenius norm. For a vector function $F: \mathbb{R}^{n_\alpha} \rightarrow \mathbb{R}^m$, $\text{SD}_{\mathcal{D}_N}(\Gamma(\alpha))$ denotes the empirical standard deviation of the elements of $\Gamma(\alpha(t))$ over a data set $\mathcal{D}_N := \{\alpha(t)\}_{t=0}^{N-1}$:

$$\text{SD}_{\mathcal{D}_N}(\Gamma(\alpha)) := [\sigma_{\mathcal{D}_N}(\Gamma_1(\alpha)) \quad \cdots \quad \sigma_{\mathcal{D}_N}(\Gamma_m(\alpha))]^\top$$

where

$$\sigma_{\mathcal{D}_N}(\Gamma_i(\alpha)) := \sqrt{\frac{1}{N} \sum_{t=0}^{N-1} (\Gamma_i(\alpha(t)) - \mathbb{E}_{\mathcal{D}_N}(\Gamma(\alpha)))^2}$$

with $\mathbb{E}_{\mathcal{D}_N}(\Gamma(\alpha)) := \frac{1}{N} \sum_{t=0}^{N-1} \Gamma_i(\alpha(t))$. A function $\alpha: \mathbb{R} \rightarrow \mathbb{R}$ is called class \mathcal{C}_1 if it is continuous and its first derivative exists.

2 Problem statement

Consider an NL system defined by the finite dimensional *state-space* (SS) representation:

$$\dot{x}(t) = f(x(t), u(t)), \quad (1a)$$

$$y(t) = h(x(t), u(t)), \quad (1b)$$

where $x: \mathbb{R} \rightarrow \mathbb{X} \subseteq \mathbb{R}^{n_x}$ is the state variable, $u: \mathbb{R} \rightarrow \mathbb{U} \subseteq \mathbb{R}^{n_u}$ is the input, and $y: \mathbb{R} \rightarrow \mathbb{Y} \subseteq \mathbb{R}^{n_y}$ is the output of the system for which it is true that (y, x, u) satisfies (1) in the ordinary sense. \mathbb{X} and \mathbb{U} are considered to be open sets containing the origin.

Assumption 1. It is assumed that the nonlinear functions f and h are factorisable as

$$f(x(t), u(t)) = \mathcal{A}(x(t), u(t))x(t) + \mathcal{B}(x(t), u(t))u(t), \quad (2a)$$

$$h(x(t), u(t)) = \mathcal{C}(x(t), u(t))x(t) + \mathcal{D}(x(t), u(t))u(t), \quad (2b)$$

where \mathcal{A} , \mathcal{B} , \mathcal{C} , and \mathcal{D} are bounded and smooth functions on $\mathbb{X} \times \mathbb{U}$. This is a mildly restrictive assumption, as a wide class of nonlinear systems can be represented by (2), such as rational or polynomial nonlinear systems.

It is supposed that for all initial condition $x_0 \in \mathbb{X}$ at any $t_0 \in \mathbb{R}$, there exists a unique solution (y, x, u) which is forward complete. Based on these assumptions, denote the solution set, i.e. the so-called behavior, of the system represented by (1) as

$$\mathbb{B}_{\text{NL}} = \{(y, x, u) \in (\mathbb{Y} \times \mathbb{X} \times \mathbb{U})^{\mathbb{R}_0^+} \mid (y, x, u) \text{ s.t. (1) holds} \\ \forall t \in \mathbb{R}_0^+ \text{ with } x \in \mathcal{C}_1^{n_x} \text{ and } x(0) = x_0 \in \mathbb{X}\}. \quad (3)$$

Introduce the notation $\mathbb{B}_{\text{NL}}^{(x)} = \{x \in \mathbb{X}^{\mathbb{R}_0^+} \mid \exists (y, u) \in (\mathbb{Y} \times \mathbb{U})^{\mathbb{R}_0^+} \text{ s.t. } (y, x, u) \in \mathbb{B}_{\text{NL}}\}$. Factorization (2) implies the following SS representation for (1):

$$\dot{x}(t) = \mathcal{A}(x(t), u(t))x(t) + \mathcal{B}(x(t), u(t))u(t), \quad (4a)$$

$$y(t) = \mathcal{C}(x(t), u(t))x(t) + \mathcal{D}(x(t), u(t))u(t). \quad (4b)$$

In this paper, three types of problems are tackled.

Problem 1 (Embedding a nonlinear model into an affine LPV model): Find an LPV-SS representation for (4) in the form of

$$\dot{x}(t) = A(\theta(t))x(t) + B(\theta(t))u(t), \quad (5a)$$

$$y(t) = C(\theta(t))x(t) + D(\theta(t))u(t), \quad (5b)$$

such that $\theta := \mu(x, u) : \mathbb{X} \times \mathbb{U} \rightarrow \Theta \subseteq \mathbb{R}^{n_\theta}$ is a bounded smooth vector function, Θ is a convex set, and matrix functions A, B, C , and D have affine dependence on $\theta(t) := [\theta_1(t) \cdots \theta_{n_\theta}(t)]^\top$, i.e.,

$$M(\theta(t)) = M_0 + \sum_{i=1}^{n_\theta} \theta_i(t)M_i, \quad (6)$$

where $M(\theta(t))$ represents any of the matrix functions $A(\theta(t)), \dots, D(\theta(t))$. Let us define the solution set of (5) as follows:

$$\begin{aligned} \mathbb{B}_{\text{LPV}} = \{ & (y, x, u, \theta) \in (\mathbb{Y} \times \mathbb{X} \times \mathbb{U} \times \Theta)^{\mathbb{R}_0^+} \mid (y, x, u, \theta) \\ & \text{s.t. (5) holds } \forall t \in \mathbb{R}_0^+ \text{ with } x \in \mathbb{C}_1^{n_x} \text{ and } x(0) = x_0 \in \mathbb{X} \}. \end{aligned} \quad (7)$$

Note that $\mathbb{B}_{\text{NL}} \subseteq \mathbb{B}_{\text{LPV}}$, due to the fact that θ can get any value in the admissible set Θ irrespective of the values of x and u . For this problem, the goal is to minimize $\mathbb{B}_{\text{LPV}} \setminus \mathbb{B}_{\text{NL}}$ in terms of a measure on $\mathbb{X} \times \mathbb{U}$ among all possible choices of A, \dots, D, μ , and Θ .

Additionally, one can alternatively seek an approximate affine LPV model as follows:

$$\dot{x}(t) = \hat{A}(\hat{\theta}(t))x(t) + \hat{B}(\hat{\theta}(t))u(t), \quad (8a)$$

$$y(t) = \hat{C}(\hat{\theta}(t))x(t) + \hat{D}(\hat{\theta}(t))u(t), \quad (8b)$$

where $\hat{\theta} := \hat{\mu}(x, u) : \mathbb{X} \times \mathbb{U} \rightarrow \hat{\Theta} \subseteq \mathbb{R}^{n_\theta}$ and $n_{\hat{\theta}} < n_\theta$ by minimizing a measure of discrepancy between the matrices $\mathcal{A}, \dots, \mathcal{D}$ and \hat{A}, \dots, \hat{D} , which measure is precisely defined later in this section.

Problem 2 (Converting an LPV model with nonlinear dependency on the scheduling variables into an affine LPV model): Given an LPV embedding of (1) in the form of

$$\dot{x}(t) = \tilde{A}(\alpha(t))x(t) + \tilde{B}(\alpha(t))u(t), \quad (9a)$$

$$y(t) = \tilde{C}(\alpha(t))x(t) + \tilde{D}(\alpha(t))u(t), \quad (9b)$$

for which $\tilde{A}, \tilde{B}, \tilde{C}$, and \tilde{D} are non-affine functions of the scheduling variable $\alpha = \kappa(x, u)$ with $\alpha(t) \in \Omega_\alpha \subset \mathbb{R}_\alpha^n$. In practice, such an embedding can easily happen in case of manual conversion of an NL model to an LPV form (e.g. in case of mechatronic systems (see Sec. 7), direct substitution of the position dependency in the inertia matrices with scheduling variables leads to LPV-SS representations with rational dependence). The goal is to find A, \dots, D, μ , and Θ in terms of (5) and (6) such that the discrepancy between the matrices $\tilde{A}, \dots, \tilde{D}$ and A, \dots, D is minimized. As in our setting LPV models represent an underlying NL system, we will consider this minimization over all realizations of α and θ in terms of $\kappa(\mathbb{B}_{\text{NL}}^{(x,u)})$ and $\hat{\mu}(\mathbb{B}_{\text{NL}}^{(x,u)})$.

Problem 3 (Obtaining an affine LPV model with reduced number of scheduling variables from an affine LPV model): Given an LPV embedding of (1) in terms of (9), where the state-space matrices are affine functions of α . The goal is to find an approximate LPV model (8) such that the discrepancy between the matrices $\tilde{A}, \dots, \tilde{D}$ and \hat{A}, \dots, \hat{D} is minimized over all realizations of α and $\hat{\theta}$ in terms of $\kappa(\mathbb{B}_{\text{NL}}^{(x,u)})$ and $\hat{\mu}(\mathbb{B}_{\text{NL}}^{(x,u)})$. This problem represents the case when the initial number of scheduling variables n_α and/or the admissible set for α are non-minimal or further reduction of these is required for feasibility of control synthesis.

Indeed, the mentioned problems are all low complexity embedding problems for either nonlinear or complex LPV systems. The objective is to find an LPV model, affine with respect to a set of constructed scheduling variables, while the accuracy and the conservativeness of the resulting embedding is taken into account.

Next we show that Problems 1-3 can be considered under a unified setting. Introduce the representation

$$\begin{bmatrix} \dot{x}(t) \\ y(t) \end{bmatrix} = L(\alpha(t)) \begin{bmatrix} x(t) \\ u(t) \end{bmatrix}, \quad (10)$$

where

$$L(\alpha(t)) := [L_{i,j}(\alpha(t))]_{m \times n} \in \mathbb{R}^{m \times n},$$

with $m := n_x + n_y$, $n := n_x + n_u$. The variables $\alpha_i(t)$ are assumed to lie in a hyper-rectangle Ω_α , which is the Cartesian product of intervals

$$\underline{\alpha}_i \leq \alpha_i(t) \leq \bar{\alpha}_i$$

where $\underline{\alpha}_i$ and $\bar{\alpha}_i$ are a priori known. Thus, $\alpha(t) \in \Omega_\alpha, \forall t \geq 0$. Then Problems 1-3 are represented as follows:

• **Problem 1:** $L(\alpha(t))$ is a NL matrix function of $\alpha(t) := [x(t)^\top \ u(t)^\top]^\top$ and is defined as follows:

$$L(\alpha(t)) := \begin{bmatrix} \mathcal{A}(x(t), u(t)) & \mathcal{B}(x(t), u(t)) \\ \mathcal{C}(x(t), u(t)) & \mathcal{D}(x(t), u(t)) \end{bmatrix}. \quad (11)$$

• **Problem 2:** $L(\alpha(t))$ is a NL matrix function of the scheduling variable $\alpha(t)$ and defined as follows:

$$L(\alpha(t)) := \begin{bmatrix} \tilde{\mathcal{A}}(\alpha(t)) & \tilde{\mathcal{B}}(\alpha(t)) \\ \tilde{\mathcal{C}}(\alpha(t)) & \tilde{\mathcal{D}}(\alpha(t)) \end{bmatrix}. \quad (12)$$

• **Problem 3:** $L(\alpha(t))$ is an affine function of scheduling variable $\alpha(t)$ and is defined as in (12).

Let us define

$$\begin{aligned} \Gamma(\alpha(t)) &:= [\Gamma_1(\alpha(t)) \ \Gamma_2(\alpha(t)) \ \cdots \ \Gamma_{n_\Gamma}(\alpha(t))]^\top \\ &= \vec{L}(\alpha(t)) \in \mathbb{R}^{n_\Gamma}, \end{aligned}$$

where $n_\Gamma = (n_x + n_y)(n_x + n_u)$. The ultimate goal in this paper is to find an LPV model with affine dependency

$$\begin{bmatrix} \dot{x}(t) \\ y(t) \end{bmatrix} = \hat{L}(\theta(t)) \begin{bmatrix} x(t) \\ u(t) \end{bmatrix}, \quad (13)$$

for (10) by introducing an affine mapping

$$\begin{aligned} \theta(t) &= [\theta_1(t) \ \cdots \ \theta_{n_\theta}(t)]^\top \\ &:= \mathcal{T}(\Gamma(\alpha(t))) \in \Omega_\theta, \quad \mathcal{T} : \mathbb{R}^{n_\Gamma} \rightarrow \mathbb{R}^{n_\theta} \end{aligned} \quad (14)$$

such that an accuracy index (defined next) is minimized for a prescribed value of $n_\theta \leq n_\Gamma$. Alternatively, the number of scheduling variables n_θ can be chosen by the designer based on the accuracy index value obtained for different number of scheduling variables. The hyper-rectangle set Ω_θ denoted by

$$\underline{\theta}_i \leq \theta_i(t) \leq \bar{\theta}_i, \quad (15)$$

is characterized by its lower and upper bounds $\underline{\theta}_i$ and $\bar{\theta}_i$. The mapping $\mathcal{T}(\Gamma(\alpha(t)))$ should be determined in such a way that the volume of Ω_θ is kept at its possible minimum to minimize the conservativeness of the obtained model.

Under the assumption that a $\mathcal{D}_N := \{\alpha(t)\}_{t=0}^{N-1}$ data set, representative w.r.t. typical operation of the system (10) (bundle of measured trajectories, equidistant gridding of $\mathbb{X} \times \mathbb{U}$, etc.) is available, we chose the accuracy index as the weighted norm:

$$\eta := \left\| W(\Pi_\alpha - \hat{\Pi}_\theta) \right\|_F \quad (16)$$

where

$$\Pi_\alpha := [\Gamma(\alpha(0)) \quad \Gamma(\alpha(T)) \quad \cdots \quad \Gamma(\alpha((N-1)T))], \quad (17a)$$

$$\hat{\Pi}_\theta := [\hat{\Gamma}(\theta(0)) \quad \hat{\Gamma}(\theta(T)) \quad \cdots \quad \hat{\Gamma}(\theta((N-1)T))], \quad (17b)$$

with $\hat{\Gamma}(\theta(t)) = \overrightarrow{\hat{L}}(\theta(t)) \in \mathbb{R}^{n_\Gamma}$, sampling time $T > 0$ and weighting

$$W := \text{diag}(\text{SD}_{\mathcal{D}_N}(\Gamma(\alpha))^{-1}). \quad (18)$$

Note that the Frobenius norm of a matrix is a convenient metric to quantify the approximation error of matrices based on *singular value decomposition* (SVD) type of projections and makes it possible to give explicit characterization of their approximation error. As it becomes clear later, our proposed method is based on such an SVD type of approximation, hence, the Frobenius norm is a natural choice of performance measure for our method. However, it is just a particular choice of norm and other matrix norms can be utilized for this purpose, although other choices lose the connection with the SVD based projection. In the subsequent sections, the problem of finding the mapping \mathcal{T} , introduced in (14), is addressed.

3 Parameter set mapping

To compute the scheduling mapping, PCA is applied on the data matrix (17a), inspired by the method in [19]. Note that contrary to the method in [19], PCA is applied on the data matrix Π_α , capturing the time-domain trajectories of all the elements of the state-space matrices, not only the scheduling trajectories. As we will show, this not only allows to jointly treat Problems 1-3, but it also allows achieving more accurate models with less number of scheduling variables compared to [19].

First, let us define the affine projection law $\mathcal{N}^{(i)}$ as follows:

$$\begin{aligned} \bar{\Gamma}_i(\alpha(t)) &:= \mathcal{N}^{(i)}(\Gamma_i(\alpha(t))) \\ &= \frac{1}{\sigma_{\mathcal{D}_N}(\Gamma_i(\alpha))} (\Gamma_i(\alpha(t)) - \mathbb{E}_{\mathcal{D}_N}(\Gamma(\alpha))) \end{aligned} \quad (19)$$

where $\sigma_{\mathcal{D}_N}(\Gamma_i(\alpha))$ and $\mathbb{E}_{\mathcal{D}_N}(\Gamma(\alpha))$ are respectively the standard deviation and the mean of $\Gamma_i(\alpha)$ over the data set \mathcal{D}_N . Now, we can define $\bar{\Gamma}(\alpha(t)) = \mathcal{N}(\Gamma(\alpha(t)))$ which means that the corresponding affine mappings are applied on the related elements of the vector $\Gamma(\alpha(t))$. Similarly, we can introduce $\bar{\Pi}_\alpha^{(i)} = \mathcal{N}^{(i)}(\Pi_\alpha^{(i)})$ and the overall $\bar{\Pi}_\alpha = \mathcal{N}(\Pi_\alpha)$ to obtain a scaled (unit variance), zero mean representation of the variation of the state-space matrices. Let

$$\bar{\Pi}_\alpha = U \Sigma V^\top \quad (20)$$

be the SVD of $\bar{\Pi}_\alpha$. Singular values indicate the principal components of the data. Small singular values indicate relatively unimportant components [29], which means that projection onto a low-dimensional subspace spanned by the dominant singular vectors will not cause losing too much information. Suppose that $\sigma_1, \sigma_2, \dots, \sigma_{n_\rho}$ are considered as the significant singular values (based on their relative magnitudes). By neglecting the singular values $\sigma_{n_\rho+1}, \dots, \sigma_{n_\Gamma}$ and partitioning U , $\Sigma := \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{n_\rho}, \dots, \sigma_{n_\Gamma})$, and V as follows:

$$\Sigma := \begin{bmatrix} \Sigma_\rho & 0 & 0 \\ 0 & \Sigma_\eta & 0 \end{bmatrix}, \quad U := [U_\rho \quad U_\eta], \quad V := [V_\rho \quad U_\eta],$$

where $\Sigma_\rho = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{n_\rho})$ and $\Sigma_\eta = \text{diag}(\sigma_{n_\rho+1}, \dots, \sigma_{n_\Gamma})$, one can obtain the following approximation for Π_α

$$\hat{\Pi}_\rho := \mathcal{N}^{-1} \left(U_\rho U_\rho^\top \bar{\Pi}_\alpha \right) = \mathcal{N}^{-1} \left(U_\rho U_\rho^\top \mathcal{N}(\Pi_\alpha) \right) \approx \Pi_\alpha, \quad (21)$$

where \mathcal{N}^{-1} denotes the rescaling and translation, respectively, such that $\mathcal{N}^{-1}(\mathcal{N}(\Pi_\alpha)) = \Pi_\alpha$.

Let us define the affine reduced mapping

$$\rho(t) := \mathcal{M}(\alpha(t)) = U_\rho^\top \bar{\Gamma}(\alpha(t)) = U_\rho^\top \mathcal{N}(\Gamma(\alpha(t))), \quad (22)$$

considering (17a) and (21), one can see that

$$\Pi_\alpha \approx \hat{\Pi}_\rho := [\hat{\Gamma}(\rho(0)) \quad \hat{\Gamma}(\rho(T)) \quad \cdots \quad \hat{\Gamma}(\rho((N-1)T))]$$

where

$$\hat{\Gamma}(\rho(t)) := \mathcal{N}^{-1}(U_\rho \rho(t)). \quad (23)$$

Subsequently, one can define

$$\hat{L}(\rho(t)) := \overleftarrow{\hat{L}}(\rho(t)). \quad (24)$$

Note that $\hat{L}(\rho(t))$ is an affine function of $\rho(t)$, and $\rho(t)$ is also an affine function of $\Gamma(\alpha(t))$. Thus, $\hat{L}(\rho(t))$ is an affine function of $\Gamma(\alpha(t))$, but depending on the function Γ (original dependencies of the state-matrices) can be a nonlinear function of α . Based on a well-known matrix approximation lemma [30], we have

$$\left\| \mathcal{N}(\Pi_\alpha) - \mathcal{N}(\hat{\Pi}_\rho) \right\|_F = \sigma_{n_\rho+1} + \cdots + \sigma_{n_\Gamma} := \eta \quad (25)$$

Bear in mind that \mathcal{N} is an affine transformation. Therefore, one can easily see that $\mathcal{N}(\Pi_\alpha) - \mathcal{N}(\Pi_\rho) = W(\Pi_\alpha - \Pi_\rho)$ where the matrix W is given by (18). This immediately implies

$$\eta = \sigma_{n_\rho+1} + \cdots + \sigma_{n_\Gamma} = \|W(\Pi_\alpha - \Pi_\rho)\|_F. \quad (26)$$

Note that $\rho(t) := [\rho_1(t) \quad \cdots \quad \rho_{n_\rho}(t)]^\top \in \mathbb{R}^{n_\rho}$ belongs to a hyper-rectangle Ω_ρ denoted by

$$\underline{\rho}_i \leq \rho_i(t) \leq \bar{\rho}_i.$$

The lower and upper bounds $\underline{\rho}_i$ and $\bar{\rho}_i$ are obtained respectively as the minimum and maximum values of $\rho_i(t)$ in terms of (22) over all admissible values of $\alpha(t) \in \Omega_\alpha$.

Taking into account that $\hat{L}(\rho(t))$ depends affinely on the newly introduced scheduling vector $\rho(t)$, which is in turn an affine vector function of the elements of $L(\alpha(t))$, one can see that the embedding Problems 1-3 have been successfully addressed. In the next section, the reduction of the associated conservativeness with the LPV models is investigated.

Remark 1. Note that the number of elements of $L(\alpha(t))$ in Problem 1 is usually greater than the number of individual nonlinear functions that appear in $L(\alpha(t))$. In this case, some of the singular values become zero. If n_ρ is chosen equal to the number of nonzero singular values, then an exact LPV model is obtained (see 6.2), and an approximate LPV model is obtained if n_ρ is selected smaller than the number of nonzero singular values. However, for Problems 2 and 3, usually an approximate affine LPV model is developed where the approximation error is characterized by the accuracy index (16).

4 Minimal enclosing hyper rectangles

The hyper rectangle set Ω_ρ contains the new scheduling variable $\rho(t)$; however, Ω_ρ is not necessarily the hyper rectangle with the smallest volume. In this section, we would like to introduce an invertible affine transformation \mathcal{R} consisting of translation and rotation as follows:

$$\theta(t) := \mathcal{R}(\rho(t)) \in \Omega_\theta, \quad \mathcal{R} : \mathbb{R}^{n_\rho} \rightarrow \mathbb{R}^{n_\theta}, \quad n_\rho = n_\theta$$

such that the hyper rectangle Ω_θ , denoted by (15), has the minimum possible volume. The problem to find the hyper rectangle Ω_θ with the minimum possible volume can be cast as the problem of finding a hyper rectangle with minimum-volume enclosing a set of points. We consider two distinct cases $n_\theta \leq 3$ and $n_\theta > 3$ for reasons that will be clear soon.

4.1 The case when $n_\theta \leq 3$

Finding the minimum volume enclosing hyper rectangle has already been tackled for two- and three-dimensional point sets in the literature [31–33]. In [34], finding a minimum-volume bounding box for a set of n points in \mathbb{R}^3 is considered where an efficient $\mathcal{O}(n + 1/\epsilon^{4.5})$ -time algorithm for computing a $(1 + \epsilon)$ -approximation of the minimum-volume bounding box is solved; thus, the running time of this algorithm is linear in n (number of the time samples). Examples of implementation of this algorithm can be found in [35, 36]. The minimum-volume bounding box obtained by any of the available methods is characterized by its vertices. Note that the bounding box is generally not aligned with the coordinate axes. To describe it with lower and upper bounds on the individual scheduling variables, which is more desirable from the viewpoint of controller synthesis, one should resort to an appropriate rotation of the scheduling space. Thus, we first find the minimum volume enclosing hyper rectangle, then, we compute a rotation of the scheduling space, i.e. transformation of $\rho(t)$, such that we align the hyper rectangle with the new scheduling coordinate axes.

To compute the rotation transformation, we take advantage of the Kabsch algorithm [37]. Let us define

$$P := [v_1 - \bar{v} \quad v_2 - \bar{v} \quad \cdots \quad v_{n_v} - \bar{v}],$$

where v_i is either in \mathbb{R}^2 or \mathbb{R}^3 and represents vertex i of the minimum-volume bounding box with n_v vertices (either $n_v = 4$ or $n_v = 8$) and $\bar{v} = \frac{1}{n_v} \sum_{i=1}^{n_v} v_i$ is the centroid of the box. Let us consider

$$Q := \begin{bmatrix} \bar{\sigma}_1 & \bar{\sigma}_1 & -\bar{\sigma}_1 & -\bar{\sigma}_1 & -\bar{\sigma}_1 & -\bar{\sigma}_1 & \bar{\sigma}_1 & \bar{\sigma}_1 \\ \bar{\sigma}_2 & -\bar{\sigma}_2 & -\bar{\sigma}_2 & \bar{\sigma}_2 & \bar{\sigma}_2 & -\bar{\sigma}_2 & -\bar{\sigma}_2 & \bar{\sigma}_2 \end{bmatrix},$$

for $n_\theta = n_\rho = 2$ scheduling variables and

$$Q := \begin{bmatrix} \bar{\sigma}_1 & \bar{\sigma}_1 & -\bar{\sigma}_1 & -\bar{\sigma}_1 & -\bar{\sigma}_1 & -\bar{\sigma}_1 & \bar{\sigma}_1 & \bar{\sigma}_1 \\ \bar{\sigma}_2 & -\bar{\sigma}_2 & -\bar{\sigma}_2 & \bar{\sigma}_2 & \bar{\sigma}_2 & -\bar{\sigma}_2 & -\bar{\sigma}_2 & \bar{\sigma}_2 \\ \bar{\sigma}_3 & \bar{\sigma}_3 & \bar{\sigma}_3 & \bar{\sigma}_3 & -\bar{\sigma}_3 & -\bar{\sigma}_3 & -\bar{\sigma}_3 & -\bar{\sigma}_3 \end{bmatrix},$$

for $n_\theta = n_\rho = 3$ scheduling variables, where $\bar{\sigma}_1, \bar{\sigma}_2$ and $\bar{\sigma}_1, \bar{\sigma}_2, \bar{\sigma}_3$ are the singular values of P for the case of $n_\theta = 2$ and $n_\theta = 3$, respectively. In terms of the Kabsch algorithm, one can obtain the transformation \mathcal{R} having been previously introduced as follows:

$$\theta(t) = \mathcal{R}(\rho(t)) = (QP^\top PQ^\top)^{\frac{1}{2}}(PQ^\top)^{-1}(\rho(t) - \bar{v}) + \bar{v}. \quad (27)$$

This way, the newly introduced scheduling variable vector $\theta(t)$ reside in a hyper rectangle Ω_θ which is defined by (15), i.e. the hyper rectangle Ω_θ can be characterized by the lower and upper bounds on the individual scheduling variables θ_i . It should be mentioned that the volume of Ω_θ is exactly equal to the volume of the initial minimum-volume bounding box due to the fact that $\mathcal{R}(\cdot)$ consists of rotation and translation only which operations preserve volume.

Remark 2. It is worth mentioning that the rotated minimum-volume bounding box, called hereafter Ω_θ , is not unique due to the fact that the rotation may be carried out in different directions by choosing alternative ordering in Q . However, irrespective of the direction of the rotation, the obtained minimum-volume bounding boxes can be characterized by the upper and lower bounds of the introduced scheduling variables. These bounding boxes can be converted to each other by changing the ordering of the scheduling variables and/or changing the sign of the variables.

4.2 The case when $n_\theta > 3$

We are not aware of any previously-published polynomial-time algorithm that tackles the problem of finding an enclosing hyper rectangle for a set of points for dimension higher than 3. In case it is required to have an LPV model with more than three scheduling variables, one solution is to find a minimum-volume ellipsoid enclosing the scheduling variable trajectories $\rho(t)$ corresponding

to the variation in \mathcal{D}_N , and then to determine a hyper rectangle enclosing this ellipsoid to be considered as the minimum-volume box. However, there is no guarantee that this hyper rectangle will be the minimum-volume hyper rectangle which encloses the possible scheduling variations, but it may potentially have a volume less than Ω_ρ , resulting in a model with reduced conservativeness.

The problem of finding the minimum-volume enclosing ellipsoid has been widely investigated in the literature, see e.g. [38–40]. An algorithm to solve this problem can be found in [41]. Suppose that the minimum-volume ellipsoid is parametrized as follows:

$$\mathcal{E} := \left\{ v \in \mathbb{R}^{n_\theta} \mid (v - \bar{v}_e)^\top P_e (v - \bar{v}_e) \leq 1 \right\},$$

where $n_\theta > 3$ and $P_e \in \mathbb{R}^{n_\theta \times n_\theta}$ is a symmetric positive-definite matrix and $\bar{v}_e \in \mathbb{R}^{n_\theta}$ is the center of the ellipsoid. In short, this ellipsoid can be obtained by solving the following convex optimization problem:

$$\begin{aligned} \min_{P_e, \bar{v}_e} \quad & -\log \det P_e, \\ \text{s.t.} \quad & (v - \bar{v}_e)^\top P_e (v - \bar{v}_e) \leq 1, \quad \forall v \in \hat{\mathcal{D}}_N, \\ & P_e > 0, \end{aligned} \quad (28)$$

where $\hat{\mathcal{D}}_N := \{\rho(t)\}_{t=0}^{N-1}$, and P_e and \bar{v}_e are decision variables. Let

$$P_e = U_e \Sigma_e V_e^\top,$$

be the singular value decomposition of P_e . Then, using the following transformation

$$\theta(t) = \mathcal{R}(\rho(t)) = U_e(\rho(t) - \bar{v}_e) + \bar{v}_e, \quad (29)$$

the minimum-volume ellipsoid is rotated around the center \bar{v}_e such that the principal axes become parallel to the coordinate axes. Now, the required hyper rectangle can be defined by the lower and upper bounds on the individual scheduling variables, which are obtained as the minimum and maximum values of θ_i .

5 Affine LPV model construction

To recapitulate the previous sections, one can obtain an approximate affine LPV model (13) with n_θ scheduling variables for any of the three considered problems by the following affine mapping

$$\theta(t) := \mathcal{T}(\alpha(t)) = \mathcal{R}(\mathcal{M}(\alpha(t))) = \mathcal{R}(U_\rho^\top \mathcal{N}(\Gamma(\alpha(t)))) \in \Omega_\theta$$

where $\mathcal{R}(\rho(t))$ is given by either (27) or (29). The hyper rectangle Ω_θ is characterized by the upper and lower bounds of θ_i .

The accuracy index for the approximate model is as follows:

$$\eta = \|W(\Pi_\alpha - \Pi_\theta)\|_F = \sigma_{n_\rho+1} + \cdots + \sigma_{n_\Gamma}, \quad (30)$$

where $\Pi_\theta = \Pi_\rho|_{\rho=\mathcal{R}^{-1}(\theta)}$. Note that $\theta(t) = \mathcal{R}(\rho(t))$ is just a change of variable; consequently, it is easy to see that $\Pi_\theta = \Pi_\rho$. Thus, (26) immediately implies (30). As we mentioned previously, $\mathcal{M}(\cdot)$ is an affine mapping; therefore, $\mathcal{T}(\cdot)$ is also an affine mapping.

Taking into account (23), (24), (27), (29) and by defining

$$\hat{\Gamma}(\theta(t)) := \mathcal{N}^{-1}(U_\rho \mathcal{R}^{-1}(\theta(t))),$$

the affine LPV model (13) is obtained where

$$\hat{L}(\theta(t)) := \hat{\Gamma}(\theta(t)).$$

Bear in mind that $\mathcal{R}(\cdot)$ and $\mathcal{N}(\cdot)$ are invertible affine transformations. Note that the number of scheduling variables n_θ can either be considered as a prescribed value or chosen based on the accuracy index η .

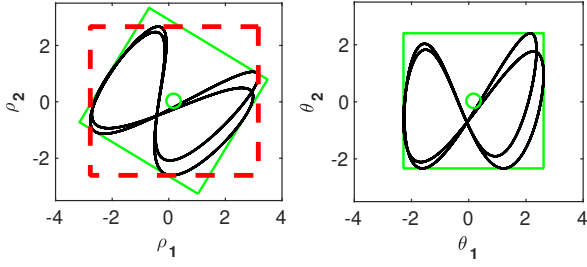


Fig. 1: Left plot: scheduling variables ρ_i (plotted over \mathcal{D}_N), Ω_ρ (red dashed box), and the minimum-volume enclosing box (green box). Right plot: scheduling variables θ_i and Ω_θ . The figure demonstrates that computation of the minimal enclosing hyper rectangle and the proposed transformation results in smaller scheduling sets and hence reduced conservativeness of the resulting model.

6 Numerical illustration

In this section, two numerical examples are provided to reveal the advantages of the proposed method in addition to comparison with some available approaches in the literature.

6.1 Example 1

This example is provided to evaluate the effectiveness of the proposed method for Problem 3 in this paper. Consider an LPV system given by (9) with the following state-space matrices

$$\begin{bmatrix} \bar{A}(\alpha) & \bar{B}(\alpha) \\ \bar{C}(\alpha) & \bar{D}(\alpha) \end{bmatrix} = \begin{bmatrix} 1 + 2\alpha_1 & 3 + \alpha_2 & 3\alpha_3 + 7\alpha_2 \\ 2 + 3\alpha_3 & 20\alpha_1 + 5\alpha_2 & 1 \\ \alpha_1 & 0 & 0 \end{bmatrix} \quad (31)$$

with scheduling variable $\alpha(t) \in [0, 2] \times [0, 5] \times [-1, 1] = \Omega_\alpha$. The goal is to obtain an approximate LPV model (13) with two scheduling variables $\theta := [\theta_1 \ \theta_2]^\top \in \Omega_\theta$ and with the minimum possible value of η , given by (16), such that Ω_θ , characterized by $\underline{\theta}_1, \theta_1, \underline{\theta}_2$, and θ_2 , has minimum-volume.

Let us consider $T = 10^{-3}$ and scheduling trajectories

$$\begin{aligned} \alpha_1(t) &= 2 \sin^2(10t), & \alpha_2(t) &= 5 \cos^2(20t + \frac{\pi}{5}), \\ \alpha_3(t) &= \sin(10t) \cos(20t). \end{aligned}$$

to generate \mathcal{D}_N with $N = 3000$. These trajectories adequately explore Ω_α and represent the typical operation of the system we would like to preserve in our LPV model. Using the proposed method, an approximate model with two scheduling variables is obtained. In the left plot of Fig. 1, the scheduling variable vector ρ , Ω_ρ , and the minimum-volume enclosing box is depicted. Additionally, the scheduling variable vector θ and Ω_θ is shown in the right plot of Fig. 1. The minimum-volume bounding box is obtained by the algorithm in [36]. The centroid of the minimum-volume enclosing box around which the rotation is carried out is $(0.1688, 0.0365)$. Ω_θ can be characterized by $-2.2798 \leq \theta_1 \leq 2.6174$ and $-2.3341 \leq \theta_2 \leq 2.4071$. The volume of Ω_ρ and Ω_θ are respectively 31.2870 and 23.2186. We have $\eta = 54.4705$ for the proposed method.

For comparison purposes, the proposed method in [19] is also applied on this system to obtain an approximate model with two scheduling variables. To visualize the results, variation of some elements of $\mathcal{A}(\alpha(t))$: $a_{11} = 1 + 2\alpha_1$, $a_{22} = 20\alpha_1 + 5\alpha_2$, and $b_{11} = 3\alpha_3 + 7\alpha_2$ and their approximate counterparts are shown in Fig. 2 for a time interval between $t = 1$ and $t = 2$. Obviously, the

proposed method provides a better approximation for the original LPV system with the same number of the scheduling variables. It is worth to mention that for the method in [19], the obtained value for η , given by (16), equals to $\eta = 68.2811$, which is greater than that of the proposed method ($\eta = 54.4705$). This also reinforces the superiority of the proposed method.

6.2 Example 2

In this example, the applicability of the proposed method for Problem 1 is numerically investigated. Consider a nonlinear system given by (4) as follows:

$$\begin{bmatrix} \mathcal{A}(x, u) & \mathcal{B}(x, u) \\ \mathcal{C}(x, u) & \mathcal{D}(x, u) \end{bmatrix} = \begin{bmatrix} 2 \sin(x_1) + 1 & 3x_1 + 5 & 0 \\ x_1 & 0 & 1 \\ \sin(x_1) & 2x_1 & 0 \end{bmatrix}. \quad (32)$$

It is assumed that $-\frac{\pi}{2} \leq x_1(t) \leq \frac{\pi}{2}$. In the proposed method, after constructing the normalized matrix $\bar{\Pi}_\alpha$ using the sampling period $T = 0.01$, one can see that the singular values are

$$\sigma_1 = 39.5533, \quad \sigma_2 = 2.3526, \quad \sigma_3 = \sigma_4 = \sigma_5 = 0,$$

which implies that using just two scheduling variables an equivalent LPV embedding of the system is available. Obviously, it is expected since all the elements of $\mathcal{E}(x(t), u(t))$ are affine functions of the terms $\sin(x_1(t))$ and $x_1(t)$ which can be considered as the scheduling variables. Note that the zero terms in $\mathcal{E}(x(t), u(t))$ are excluded from construction of $\Gamma(\alpha(t))$ without any loss of generality. Applying the proposed method and considering two scheduling variables yields an exact LPV model (13) with $\hat{L}(\theta(t))$ as

$$\begin{bmatrix} 0.6337\theta_1 + 0.7773\theta_2 + 1 & 1.2226\theta_1 - 0.9968\theta_2 + 5 & 0 \\ 0.4075\theta_1 - 0.3323\theta_2 & 0 & 1 \\ 0.3169\theta_1 + 0.3887\theta_2 & 0.8151\theta_1 - 0.6645\theta_2 & 0 \end{bmatrix}, \quad (33)$$

where the scheduling variables θ_1 and θ_2 are defined by the following affine mapping of the elements of (32):

$$\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 0.3150 & 0.3864 \\ 0.1638 & -0.1335 \\ 0.4913 & -0.4006 \\ 0.6301 & 0.7728 \\ 0.2457 & -0.2003 \end{bmatrix}^\top \begin{bmatrix} 2 \sin(x_1) + 1 \\ 3x_1 + 5 \\ x_1 \\ \sin(x_1) \\ 2x_1 \end{bmatrix} + \begin{bmatrix} -1.1339 \\ 0.2812 \end{bmatrix}.$$

The above given construction implies the scheduling map

$$\mu(x(t)) = \begin{cases} \theta_1(t) = 1.2601 \sin(x_1(t)) + 1.4740x_1(t), \\ \theta_2(t) = 1.5456 \sin(x_1(t)) - 1.2017x_1(t). \end{cases}$$

If $\theta_1(t)$ and $\theta_2(t)$ are substituted with the scheduling map μ in $\hat{L}(\theta(t))$, then the same nonlinear model (32) is obtained. However, note that the proposed method is an automated affine LPV embedding approach for the nonlinear systems.

One can see that σ_2 is negligible in comparison with σ_1 . Therefore, it is also possible to obtain an approximately accurate LPV model with just one scheduling variable for this system. After applying the proposed method, an affine LPV model is obtained:

$$\hat{L}(\theta) = \begin{bmatrix} 0.6337\theta_1 + 1 & 1.2226\theta_1 + 5 & 0 \\ 0.4075\theta_1 & 0 & 1 \\ 0.3169\theta_1 & 0.8151\theta_1 & 0 \end{bmatrix}, \quad (34)$$

with $\theta_1(t) = \mu(x(t)) = 1.2601 \sin(x_1(t)) + 1.4740x_1(t)$.

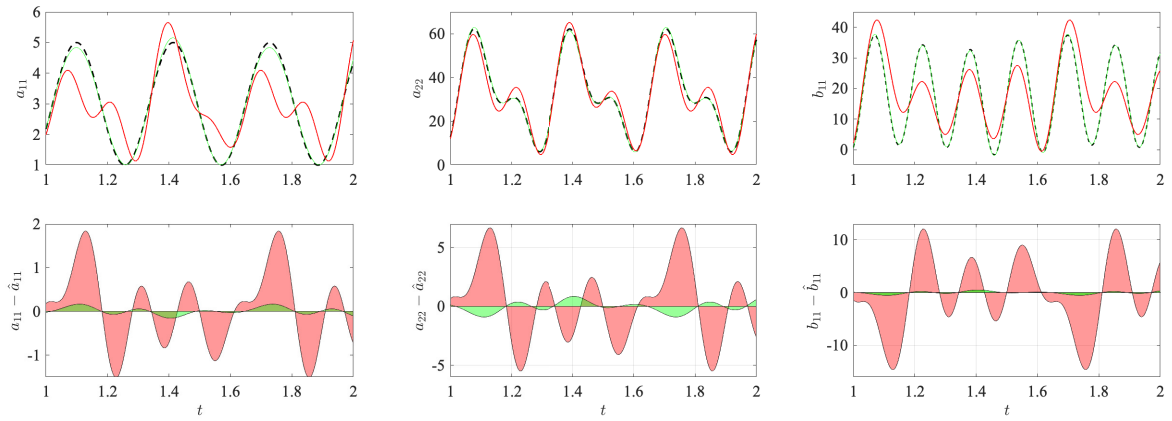


Fig. 2: Dashed lines: true trajectories of a_{11} , a_{22} , and b_{11} . Thick lines (red): approximate trajectories obtained by the method of [19]. Thin lines (green): approximate trajectories obtained by the proposed method. The figures show that the proposed method achieves significantly lower approximation error of the system with 2 scheduling variables than the method in [19].

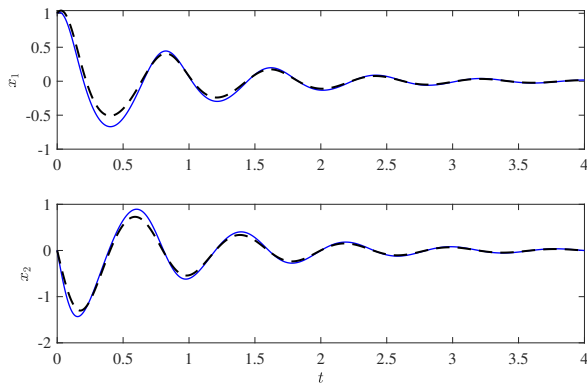


Fig. 3: The state trajectories of the LPV model (blue lines) and the nonlinear system (dashed black lines) in a time-domain simulation using a gain-scheduled state feedback controller for Example 6.2. The responses show that the LPV model with a single scheduling variable has highly similar closed-loop response as the nonlinear system.

As these systems are unstable, for comparison purposes, the response of the nonlinear and the affine LPV system (with one scheduling variable) are computed under the same gain-scheduled state feedback controller. Fig. 3 reveals good coincidence between the time-domain simulation results starting from the initial state $x(0) = [1 \ 0]^T$.

7 Experimental example

In this section, the proposed method is applied on a 3-DOF gyroscope system by Quanser, shown in Fig. 4. Using a first-principle model of the system and measured data, an LPV model of the system is constructed with our method. Subsequently, exploiting the obtained model, a full-order gain-scheduled output feedback controller is designed and applied on the setup. Converting the motion model of the gyroscope to an LPV form is challenging and results in an excessive number of scheduling variables [42], so obtaining an LPV model with low number of scheduling variables is an achievement in itself by the proposed method.



Fig. 4: The gyroscope experimental setup by Quanser.

7.1 Plant description

The gyroscope consists of a golden flywheel mounted inside an inner blue gimbal which in turn is mounted inside an outer red gimbal. The red gimbal is attached to a rotating silver frame. In the experiment considered in this paper, it is supposed that the rectangular silver frame is fixed. The blue and red gimbals can be actuated about their rotation axes using DC motors and the angular position of both gimbals are measured using optical encoders. The flywheel is actuated using another motor.

In our experiment, the flywheel is regulated by a controller which follows an unknown reference signal. The objective is to achieve servo control of the red and blue frames together by rejecting the disturbance generated by the change of the velocity of the flywheel. Let $q_2(t)$ and $q_3(t)$ represent the angular position of the blue and red gimbals, respectively. Moreover, the angular velocity of the flywheel, the blue gimbal, and the red gimbal are denoted by $\dot{q}_1(t)$, $\dot{q}_2(t)$, and $\dot{q}_3(t)$. The torque applied on the blue and red gimbals are given by $\tau_b(t)$ and $\tau_r(t)$.

7.2 LPV modeling

Through the Euler-Lagrange equations and frames defined for the rational bodies in the gyroscope, a dynamical motion model of the system is derived:

$$\mathcal{M}(q)\ddot{q} + \mathcal{C}(q, \dot{q})\dot{q} = \tau. \quad (35)$$

where $\mathcal{M}(q)$ is the inertia matrix and $\mathcal{C}(q, \dot{q})$ is the Coriolis matrix whose elements are derived as the result of the summation of the Christoffel symbols and the generalized angular velocities. The

Table 1 RMSE values of the obtained closed-loop simulation response of the LPV model w.r.t. the nonlinear gyroscope model operated by the same controller.

	τ_2	τ_3	q_2	q_3
RMSE	0.0036	0.0049	0.0013	0.0029

corresponding coefficients of these matrix functions are identified using measured data from the physical system and prediction error minimization with nonlinear optimization. As the flywheel can be seen as a separate subsystem, an NL-SS model of (35) w.r.t. the dynamics of the blue and red gimbals is derived in terms of (2) with

$$\begin{bmatrix} \mathcal{A}(x, q_1) & \mathcal{B}(x) \\ \mathcal{C} & \mathcal{D} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & f_1(x, q_1) & f_2(x, q_1) & g_1(x) & g_2(x) \\ 0 & 0 & f_3(x, q_1) & f_4(x, q_1) & g_3(x) & g_4(x) \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (36)$$

where $x := [q_2 \ q_3 \ \dot{q}_2 \ \dot{q}_3]^\top$ and $u = [\tau_b \ \tau_r]^\top$. Due to the division by $\mathcal{M}(q)$ to derive (36), the rather complicated rational trigonometrical expressions of f_1, \dots, f_4 , which all depend on $(q_2, q_3, \dot{q}_1, \dot{q}_2, \dot{q}_3)$, and also g_1, \dots, g_4 , which depend on (q_2, q_3) , are not given here. The interested reader can obtain them from the equations provided in [43]. (36) is a difficult nonlinear model for which we would like to obtain a low complexity affine LPV model (Problem 1). Using the proposed method with $n_\theta = 2$, an affine LPV model is obtained:

$$\dot{x}_p(t) = A_p(\theta(t))x_p(t) + B_p(\theta(t))u(t), \quad (37)$$

$$y(t) = C_p x_p(t), \quad (38)$$

where $\theta := [\theta_1 \ \theta_2]^\top$. The corresponding matrices in the affine $A_p(\theta)$, $B_p(\theta)$, and the constant matrix C_p are not included here for the sake of brevity. Moreover, the following bounds on the scheduling variables are obtained:

$$-4.5577 \leq \theta_1(t) \leq 2.3802, \quad -3.4307 \leq \theta_2(t) \leq 3.4885.$$

Additionally, one can readily compute the bounds on the derivative of the scheduling variables given below which are required for the controller synthesis in the next sections.

$$-70.9504 \leq \dot{\theta}_1(t) \leq 81.4165, \quad -16.9968 \leq \dot{\theta}_2(t) \leq 33.9214.$$

Thus, $\dot{\theta}(t)$ also lies in a hyper rectangle $\dot{\theta}(t) \in \Lambda_\theta$. The quality of the obtained LPV model is assessed in a closed-loop simulation study with an LPV controller designed in Section 7.3 and compared with the closed loop response of the nonlinear system operated with the same controller. In case of the nonlinear gyroscope model, the angular velocity of the disc \dot{q}_1 is regulated by a second controller to track a sinusoidal reference. Two multisine signals, different from those employed to obtain \mathcal{D}_N and to construct Π_α , are used as the desired reference signals for q_2 and q_3 in the closed-loop simulation. The related results for τ_b, τ_r, q_2 , and q_3 are shown in Fig. 5. The *root mean square error* (RMSE) of the obtained response of the LPV model w.r.t. the true system response is given in Table 1. Obviously, the LPV model well captures the dynamics of the original system. Moreover, magnitude plots of the frequency response of the LPV model are depicted in Fig. 7 for some frozen scheduling variables in the related intervals which clearly reveal the significant variation of the model over the scheduling set.

7.3 Control synthesis

In this section, using the obtained LPV model in Section 7.2, a gain-scheduled full-order output feedback controller is sought for the gyroscope based on the closed-loop setup shown in Fig. 6. Roughly

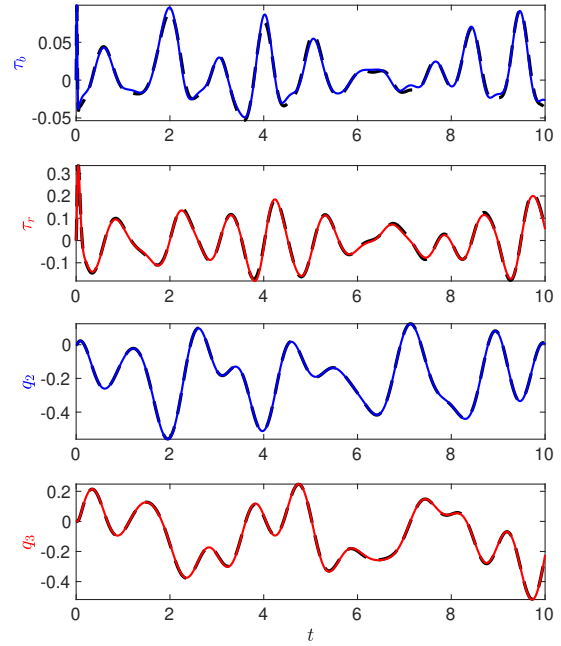


Fig. 5: Closed-loop validation results in terms of the requested input torques τ_b, τ_r and resulting angular positions q_2, q_3 of the original nonlinear system (solid-red lines for gimbal red and solid-blue lines for gimbal blue) and the LPV model (dashed-black line).

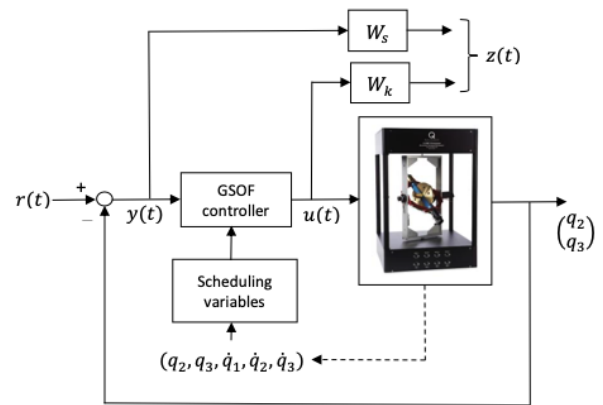


Fig. 6: Closed-loop control setup for the gyroscope.

speaking, the H_∞ -type performance, more precisely the induced \mathcal{L}_2 -gain performance, is considered to shape the frozen sensitivity and control sensitivity functions of the feedback system. To obtain good tracking performance, while the actuator constraints are taken into account, the following weighting functions are employed:

$$W_s = \begin{bmatrix} \frac{32s+20000}{1.6s+1} & 0 \\ 0 & \frac{11.2s+7000}{1.6s+1} \end{bmatrix},$$

$$W_k = \begin{bmatrix} \frac{0.024s+1.5}{0.0016s+1} & 0 \\ 0 & \frac{0.096s+6}{0.0016s+1} \end{bmatrix}.$$

The weighting function W_s has been designed based on the standard mixed-sensitivity shaping approach to shape the frozen sensitivity frequency responses of the closed-loop system. W_s is chosen such

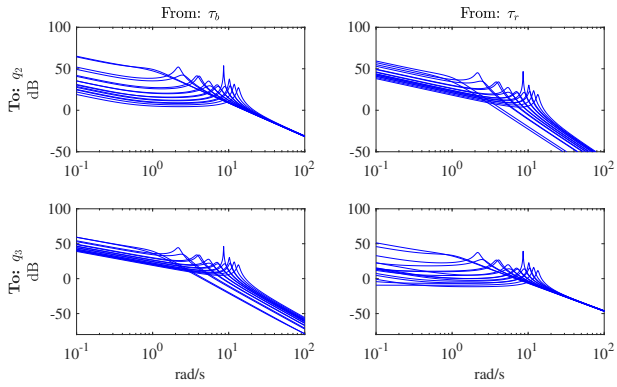


Fig. 7: Magnitude plots of the open-loop frequency responses of the LPV model for frozen (constant) scheduling variables, where significant gain and pole variations can be observed over the scheduling range.

that corresponds to a low-pass filter with bandwidth 0.7 rad/s and low frequency gain of about 80 dB to ensure good disturbance attenuation, fast response time and less than 20% of expected overshoot. Similarly the control sensitivity weighting function, W_k , has been chosen considering the amplitude and frequency constraints on the voltage to be applied to the DC motors of the experimental setup. The open-loop weighted plant displayed in Fig. 6 (by removing the controller) can be expressed as:

$$\begin{aligned} \dot{x}(t) &= A(\theta(t))x(t) + B_r(\theta(t))r(t) + B_u(\theta(t))u(t), \\ z(t) &= C_z(\theta(t))x(t) + D_r(\theta(t))r(t) + D_u(\theta(t))u(t), \\ y(t) &= C_y(\theta(t))x(t) + D_y(\theta(t))r(t), \end{aligned} \quad (39)$$

where $x(t) \in \mathbb{R}^n$, $r(t) \in \mathbb{R}^m$, $u(t) \in \mathbb{R}^p$, $z(t) \in \mathbb{R}^q$, and $y(t) \in \mathbb{R}^r$ respectively denote the state vector, the external input, the control input, the performance output, and the measured output of the system, with

$$\begin{bmatrix} A(\theta(t)) & B_r(\theta(t)) & B_u(\theta(t)) \\ C_z(\theta(t)) & D_r(\theta(t)) & D_u(\theta(t)) \\ C_y(\theta(t)) & D_y(\theta(t)) & 0 \end{bmatrix} = \begin{bmatrix} A_p(\theta(t)) & 0 & 0 & 0 & B_p(\theta(t)) \\ -B_s C_p(\theta(t)) & A_s & 0 & B_s & 0 \\ 0 & 0 & A_k & 0 & B_k \\ -D_s C_p(\theta(t)) & C_s & 0 & D_s & 0 \\ 0 & 0 & C_k & 0 & D_k \\ -C_p(\theta(t)) & 0 & 0 & I & 0 \end{bmatrix}, \quad (40)$$

where (A_s, B_s, C_s, D_s) and (A_k, B_k, C_k, D_k) are the state-space realizations of W_s and W_k , respectively.

The goal is to design the full-order gain-scheduled controller

$$\mathcal{K}(\theta) : \begin{cases} \dot{x}_c(t) = A_c(\theta(t))x_c(t) + B_c(\theta(t))y(t), \\ u(t) = C_c(\theta(t))x_c(t) + D_c(\theta(t))y(t), \end{cases}$$

such that it stabilizes the closed-loop system and assures an upper bound γ on the induced \mathcal{L}_2 -gain performance. To design the controller, we use the method detailed in [44].

The resulting parameter-dependent LMI problems are solved through finite-dimensional LMI relaxation exploiting homogeneous polynomial matrices inspired by the method of [45]. To this end, YALMIP [46] and ROLMIP [47] interfaces for the LMI solver MOSEK [48] are employed. With a $\lambda = 0.001$ an 8th-order controller is obtained with a guaranteed \mathcal{L}_2 -gain performance bound of 240.81.

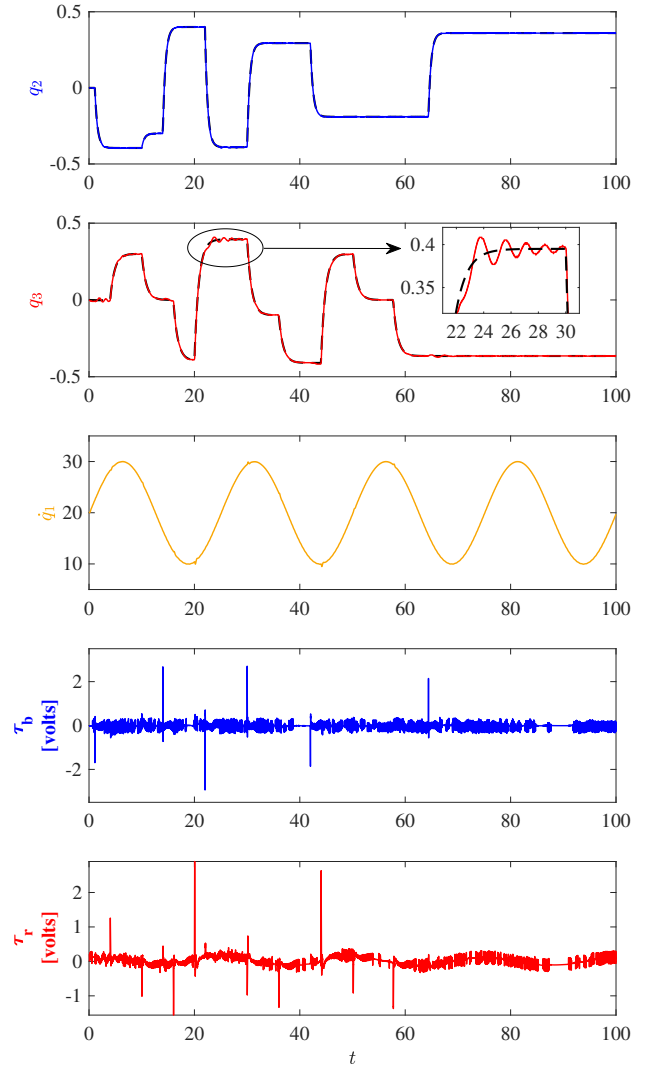


Fig. 8: Experimental results of the designed LPV controller with the gyroscope. Dashed lines: the desired reference. Solid lines: the obtained results from the real setup. The LPV controller designed based on the low complexity LPV embedding of the plant using the proposed approach of the paper shows adequate tracking performance with this highly nonlinear system.

7.4 Experimental results

In this section, the gain-scheduled controller designed based on the LPV model obtained via the proposed method in this paper is experimentally validated on the laboratory setup. The controller is described in block diagrams in MATLAB/Simulink. Then using a dSPACE board that implements a real time interface, it is applied on the setup. The angular velocity of the flywheel is made to track a sinusoidal reference between 10 to 30 rad/sec. Since the movements of blue and red gimbals do not considerably affect the rotational velocity of the flywheel, a simple proportional controller is employed to ensure the tracking. The change of the velocity of the flywheel is considered as the exogenous disturbance, and have to be rejected by the controller. The reference trajectories for gimbals blue and red, i.e. q_2 and q_3 , change within ± 0.4 rad and are designed

to cover different relative angular positions of the blue and the red gimbals. The desired and actual positions of the gimbals, the angular velocity of the flywheel, and the control input signals are shown in Fig. 8. One can see that both gimbals track successfully the related reference trajectories while the input torques τ_b and τ_r remain within acceptable levels.

8 Conclusion

A novel method taking advantage of PCA is devised in this paper for LPV embedding of nonlinear systems. Contrary to the available methods for LPV embedding of nonlinear models based on PCA, the PCA in this paper is applied on a data matrix consisting of the trajectories of all elements of the state-space matrices not solely the scheduling variables. Furthermore, by finding the minimum bounding box of the scheduling variables and its proper rotation, the conservativeness related to over bounding the admissible region of the scheduling variables is reduced by introducing an invertible transformation to find a new set of scheduling variables. In addition to academic examples, the proposed method is deployed to generate an LPV model for a 3-DOF gyroscope. Assessment of the LPV model and evaluation of the closed-loop performance, which is obtained by a designed gain-scheduled controller exploiting the developed LPV model, successfully demonstrate the applicability of the developed method in practice.

Acknowledgment

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement nr. 714663).

9 References

- W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401–1425, 2000.
- C. Hoffmann and H. Werner, "A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 416–433, March 2015.
- R. Toth, *Modeling and identification of linear parameter-varying systems*, ser. Lecture notes in control and information sciences. Germany: Springer, 2010.
- R. Tóth, V. Laurain, M. Gilson, and H. Garnier, "Instrumental variable scheme for closed-loop LPV model identification," *Automatica*, vol. 48, no. 9, pp. 2314–2320, 2012.
- J. Goos and R. Pintelon, "Continuous-time identification of periodically parameter-varying state space models," *Automatica*, vol. 71, pp. 254–263, 2016.
- V. Laurain, R. Tóth, W.-X. Zheng, and M. Gilson, "Nonparametric identification of LPV models under general noise conditions: An ls-svm based approach," *IFAC Proceedings Volumes*, vol. 45, no. 16, pp. 1761–1766, 2012, 16th IFAC Symposium on System Identification. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1474667015382112>
- R. Tóth, P. S. C. Heuberger, and P. M. J. Van den Hof, *Prediction-Error Identification of LPV Systems: Present and Beyond*. Boston, MA: Springer US, 2012, pp. 27–58.
- Y. Zhao, B. Huang, H. Su, and J. Chu, "Prediction error method for identification of LPV models," *Journal of Process Control*, vol. 22, no. 1, pp. 180–193, 2012.
- A. A. Bachnas, R. Tóth, J. H. A. Ludlage, and A. Mesbah, "A review on data-driven linear parameter-varying modeling approaches: A high-purity distillation column case study," *Journal of Process Control*, vol. 24, no. 4, pp. 272–285, 2014.
- X. Liu, X. Yang, P. Zhu, and Y. Wang, "Robust multimodel identification of LPV systems with missing observations based on t-distribution," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–7, 2019.
- J. Y. Shin, G. Balas, and M. A. Kaya, "Blending methodology of linear parameter varying control synthesis of F-16 aircraft system," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 6, pp. 1040–1048, 2002.
- A. Marcos and G. J. Balas, "Development of linear-parameter-varying models for aircraft," *Journal of Guidance, Control and Dynamics*, vol. 27, no. 2, pp. 218–228, 2004.
- A. Zin, O. Sename, P. Gáspár, L. Dugard, and J. Bokor, "An LPV/ H_∞ active suspension control for global chassis technology: Design and performance analysis," in *Proc. of the American Control Conference*, 2006, pp. 2945–2950.
- W. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401–1425, 2000.
- D. J. Leith and W. E. Leithhead, "Gain-scheduled nonlinear systems: Dynamic analysis by velocity based linearization families," *Int. Journal of Control*, vol. 70, pp. 289–317, 1998.
- M. Schoukens and R. Tóth, "Linear parameter varying representation of a class of mimo nonlinear systems," *IFAC-PapersOnLine*, vol. 51, no. 26, pp. 94–99, 2018, 2nd IFAC Workshop on Linear Parameter Varying Systems LPVS 2018.
- R. Robles, A. Sala, and M. Bernal, "Performance-oriented quasi-LPV modeling of nonlinear systems," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 5, pp. 1230–1248, 2019.
- H. Abbas, R. Tóth, M. Petreczky, N. Meskin, and J. Mohammadpour, "Embedding of nonlinear systems in a linear parameter-varying representation," in *Proceedings of the 19th IFAC World Congress, 24-29 August 2014, Cape Town, South Africa*, 2014, pp. 6907–6913.
- A. Kwiatkowski and H. Werner, "PCA-based parameter set mappings for LPV models with fewer parameters and less overbounding," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 4, pp. 781–788, July 2008.
- C. Hoffmann and H. Werner, "Compact LFT-LPV modeling with automated parameterization for efficient LPV controller synthesis," in *2015 American Control Conference (ACC)*, July 2015, pp. 119–124.
- M. M. Siraj, R. Tóth, and S. Weiland, "Joint order and dependency reduction for LPV state-space models," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, Dec 2012, pp. 6291–6296.
- C. W. Scherer, "LPV control and full block multipliers," *Automatica*, vol. 37, no. 3, pp. 361–375, 2001.
- A. Hjartarson, P. Seiler, and A. Packard, "LPVTools: A toolbox for modeling, analysis, and synthesis of parameter varying control systems," in *Proc. of the 1st IFAC Workshop on Linear Parameter Varying Systems*, Grenoble, France, Oct. 2015, pp. 139–145.
- J. D. Caigny, J. F. Camino, R. C. L. F. Oliveira, P. L. D. Peres, and J. Swevers, "Gain-scheduled H_∞ control of discrete-time polytopic time-varying systems," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, Dec 2008, pp. 3872–3877.
- J. Daafouz, J. Bernussou, and J. C. Geromel, "On Inexact LPV Control Design of Continuous-Time Polytopic Systems," *IEEE Transactions on Automatic Control*, vol. 53, no. 7, pp. 1674–1678, Aug 2008.
- M. Sato and D. Peaucelle, "Gain-scheduled output-feedback controllers using inexact scheduling parameters for continuous-time LPV systems," *Automatica*, vol. 49, no. 4, pp. 1019–1025, 2013.
- A. Sadeghzadeh, "LMI relaxations for robust gain-scheduled control of uncertain linear parameter varying systems," *IET Control Theory Applications*, vol. 13, no. 4, pp. 486–495, 2019.
- J. Hanema, M. Lazar, and R. Tóth, "Heterogeneously parameterized tube model predictive control for LPV systems," *Automatica*, vol. 111, pp. 1–13, 2019.
- P. J. Oliver and C. Shakiban, *Applied Linear Algebra*. Englewood Cliffs, NY: Prentice-Hall, 2006.
- C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.
- H. Freeman and R. Shapira, "Determining the minimum-area enclosing rectangle for an arbitrary closed curve," *Commun. ACM*, vol. 18, no. 7, pp. 409–413, Jul. 1975.
- G. Toussaint, "Solving geometric problems with the rotating calipers," in *IEEE Mediterranean Electrical Conference*, 1983.
- J. O'Rourke, "Finding minimal enclosing boxes," *International Journal of Computer & Information Sciences*, vol. 14, no. 3, pp. 183–199, Jun 1985.
- G. Barequet and S. Har-Peled, "Efficiently Approximating the Minimum-Volume Bounding Box of a Point Set in Three Dimensions," *Journal of Algorithms*, vol. 38, no. 1, pp. 91–109, 2001.
- J. Korsawe, "Minimal bounding box," 2020. [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/18264-minimal-bounding-box>
- J. Diener, "2D minimal bounding box," 2020. [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/31126-2d-minimal-bounding-box>
- W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A*, vol. 32, no. 5, pp. 922–923, Sep 1976.
- L. G. Khachiyan, "Rounding of polytopes in the real number model of computation," *Mathematics of Operations Research*, vol. 21, no. 2, pp. 307–320, 1996.
- P. Kumar and E. A. Yildirim, "Minimum-Volume Enclosing Ellipsoids and Core Sets," *Journal of Optimization Theory and Applications*, vol. 126, no. 1, pp. 1–21, 2005.
- M. J. Todd and E. A. Yildirim, "On Khachiyan's algorithm for the computation of minimum-volume enclosing ellipsoids," *Discrete Applied Mathematics*, vol. 155, no. 13, pp. 1731–1744, 2007.
- N. Moshtagh, "Minimum volume enclosing ellipsoid," 2020. [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/9542-minimum-volume-enclosing-ellipsoid>
- C. Hoffmann and H. Werner, "LFT-LPV modeling and control of a control moment gyroscope," in *Proc. of the 54th IEEE Conference on Decision and Control*, Osaka, Japan, Dec. 2015, pp. 5328–5333.
- T. Bloemers and R. Tóth, "Equations of motion of a control moment gyroscope," Eindhoven University of Technology, Tech. Rep., 2019.
- A. Sadeghzadeh, "Gain-scheduled continuous-time control using polytope-bounded inexact scheduling parameters," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 17, pp. 5557–5574, 2018.
- R. C. L. F. Oliveira and P. L. D. Peres, "Parameter-Dependent LMIs in Robust Analysis: Characterization of Homogeneous Polynomially Parameter-Dependent Solutions Via LMI Relaxations," *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1334–1340, Jul 2007.
- J. Löfberg, "YALMIP: A Toolbox for Modeling and Optimization in MATLAB," in *CACSD Conference*, 2004.
- C. M. Agulhari, R. C. L. F. de Oliveira, and P. L. D. Peres, "Robust LMI Parser: a computational package to construct LMI conditions for uncertain systems," in *19th Brazilian Conference on Automation*, Campina Grande, PB, Brazil, 2012, pp. 2298–2305.

48 MOSEK ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28)*, 2015. [Online]. Available: <http://docs.mosek.com/7.1/toolbox/index.html>