Received: 7 January 2018 | Revised: 25 August 2021 | Accepted: 26 August 2021

DOI: 10.1002/rnc.5799

RESEARCH ARTICLE

WILEY

LPV modeling of nonlinear systems: A multi-path feedback linearization approach

Hossam S. Abbas[1,2] | **Roland Tóth**[3,4] | **Mihály Petreczky**[5] | **Nader Meskin**[6] | **Javad Mohammadpour Velni**[7] | **Patrick J.W. Koelewijn**[3]

[1]Institute for Electrical Engineering in Medicine, University of Lübeck, Lübeck, Germany

[2]Electrical Engineering Department, Faculty of Engineering, Assiut University, Assiut, Egypt

[3]Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

[4]Systems and Control Laboratory, Institute for Computer Science and Control, Budapest, Hungary

[5]Centre de Recherche en Informatique, Signal et Automatique de Lille, Villeneuve-d'Ascq, France

[6]Department of Electrical Engineering, College of Engineering, Qatar University, Doha, Qatar

[7]School of Electrical and Computer Engineering, University of Georgia, Athens, Georgia, USA

Correspondence
Roland Tóth, Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB, Eindhoven, The Netherlands.
Email: r.toth@tue.nl

Funding information
Deutsche Forschungsgemeinschaft, Grant/Award Number: 419290163; H2020 European Research Council, Grant/Award Number: 714663; National Science Foundation, Grant/Award Number: 1762595

Abstract

This article introduces a systematic approach to synthesize linear parameter-varying (LPV) representations of nonlinear (NL) systems which are described by input affine state-space (SS) representations. The conversion approach results in LPV-SS representations in the observable canonical form. Based on the relative degree concept, first the SS description of a given NL representation is transformed to a normal form. In the SISO case, all nonlinearities of the original system are embedded into one NL function, which is factorized, based on a proposed algorithm, to construct an LPV representation of the original NL system. The overall procedure yields an LPV model in which the scheduling variable depends on the inputs and outputs of the system and their derivatives, achieving a practically applicable transformation of the model in case of low order derivatives. In addition, if the states of the NL model can be measured or estimated, then a modified procedure is proposed to provide LPV models scheduled by these states. Examples are included to demonstrate both approaches.

KEYWORDS

behavioral approach, dynamic dependence, equivalence transformation, linear parameter-varying systems

1 | INTRODUCTION

The *linear parameter-varying* (LPV) framework was introduced to address the control of *nonlinear* (NL) and *time-varying* (TV) systems using the extensions of powerful *linear time-invariant* (LTI) approaches such as $\mathcal{H}_2/\mathcal{H}_\infty$ optimal control and model predictive control, see for example, References 1-5. LPV systems are dynamic models capable of describing NL/TV behaviors in terms of a linear structure. Signal relations between the inputs and outputs in an LPV representation are assumed to be linear, but, at the same time, dependent on a so-called *scheduling variable* p (n_p-dimensional signal), which is assumed to be measurable and free (external) in the modeled system and taking values from a so-called *scheduling region* $\mathbb{P} \subseteq \mathbb{R}^{n_\mathrm{p}}$, often restricted to be a compact set. In this way, variation of p represents time-variance, changing operating conditions, and so forth, and aims at the embedding of the original NL/TV behavior into the solution set of an LPV system representation.[6,7] While the former objective is pursued by the so-called *global* LPV modeling approaches, alternatively, one can aim at the approximation of the NL/TV behavior by the interpolation of various linearizations of the system around operating points or signal trajectories, often referred to as *local* modeling, see, for example, References 8-10.

For the global modeling methodology we intend to investigate in this article, it is important to shed light on the often vaguely defined concept of LPV embedding. Assume that a continuous-time system $\mathcal{G}$, depicted in Figure 1A, is given which describes the (possibly nonlinear) dynamical relation between the signals $w : \mathbb{R} \to \mathbb{W}$, where $\mathbb{W}$ is a given set. For example consider the forced *Van der Pol* equation:[11]

$$\dot{x}_1 = x_2, \tag{1a}$$

$$\dot{x}_2 = -x_1 + \alpha\left(1 - x_1^2\right)x_2 + u, \tag{1b}$$

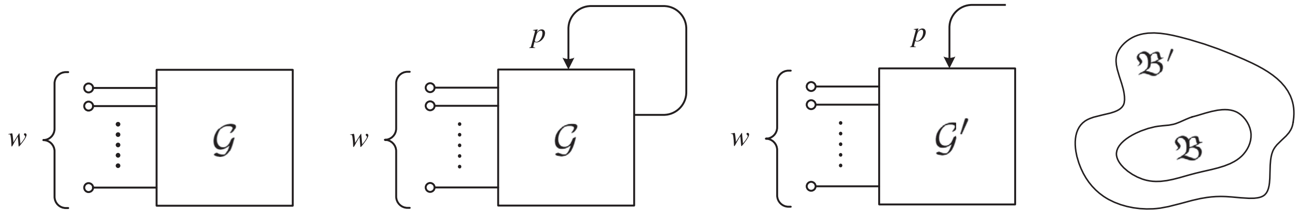
$$y = x_1, \tag{1c}$$

where, $\begin{bmatrix} x_1 & x_2 \end{bmatrix}^\top : \mathbb{R} \to \mathbb{R}^2$ is the state variable, while $w = \begin{bmatrix} u & y \end{bmatrix}^\top$ are the inputs and outputs of the system with $\mathbb{W} = \mathbb{R}^2$. Let $\mathfrak{B} \subseteq \mathbb{W}^\mathbb{R}$ ($\mathbb{W}^\mathbb{R}$ stands for all maps from $\mathbb{R}$ to $\mathbb{W}$) containing all trajectories of w that are compatible with $\mathcal{G}$, that is, they are solutions of (1). We call $\mathfrak{B}$ the (manifest) behavior of the system $\mathcal{G}$. A common practice in LPV modeling is to introduce an auxiliary variable p, with range $\mathbb{P}$, and reformulate $\mathcal{G}$ as shown in Figure 1B, where it holds true that if the loop is disconnected and p is assumed to be a known signal as in Figure 1C, then the "remaining" relations of w are linear. This can be achieved in (1) by taking, as a possible choice, $p = x_1 = y$:

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \left[\begin{array}{cc|c} 0 & 1 & 0 \\ -1 & \alpha(1-p^2) & 1 \\ \hline 1 & 0 & 0 \end{array}\right] \begin{bmatrix} x \\ u \end{bmatrix}. \tag{2}$$

Applying this reformulation with a disconnected p and assuming that all trajectories of p are allowed, that is, p is a free variable with $p \in \mathbb{P}^\mathbb{R}$ independent of y, the possible trajectories of this reformulated system $\mathcal{G}'$ form a solution set of (2), denoted as $\mathfrak{B}'$, which contains $\mathfrak{B}$ as visualized in Figure 1D. This concept of formulating $\mathcal{G}'$, a linear, but p-dependent description of $\mathcal{G}$, enables the use of simple stability analysis and convex controller synthesis, see for example, References 1-3, which can be conservative w.r.t. $\mathcal{G}$, but computationally more attractive and robust than other approaches directly addressing $\mathfrak{B}$. Control synthesis based on the above mentioned modeling procedure results in the implementation of an LPV controller $\mathcal{K}$ visualized in Figure 2. It is obvious that a key assumption is that p must be "observable" from the real system. The observed value of p is required to complete the hidden relation of p to the other variables in (2) and enable a linear controller to schedule its behavior according to p to regulate (1). Hence, this can be seen as a multi-path feedback linearization, similar to the well-known approach in NL system theory, see Reference 12, as the obtained information from the system in terms of p is fed back to arrive to a varying linear relation (2) (in contrast with the NL theory where the resulting behavior is intended to be LTI).

Following the above procedure, the scheduling variable p itself can appear in many different relations w.r.t. the original variables w. If p is a free variable w.r.t. $\mathcal{G}$, for example, wind speed for a wind turbine,[13] then we can speak about a *true parameter-varying system* without conservativeness. However, in many practical applications, like in our example, it happens that p depends on other signals, like inputs, outputs, or states of the modeled system (e.g., operating conditions). Such situations are often warningly labeled to be *quasi*-LPV (q-LPV). Based on the toy example (2), what really happens in



(A) Original plant. (B) Characterization of p. (C) LPV form by disconnecting p. (D) Relation of the resulting behaviors.

FIGURE 1 The concept of LPV modeling

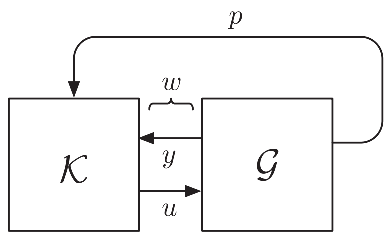


FIGURE 2 The concept of LPV control

those cases is that the assumed freedom of p only introduces conservativeness in the embedding of the nonlinear behavior. Hence, one important objective of LPV modeling, besides achieving complete embedding, is to *minimize* such *conservativeness*. Furthermore, it is often tempting to choose state variables as p that are hardly measurable or cannot be reliably estimated from the measurements. For example, in (1), we could have chosen $p = x_1x_2$ which is not directly measurable. Such choices can result in a loss of internal stability of the closed-loop system, as an uncontrollable/unobservable mode can be introduced between the observer used to track p and the controller that schedules based on it. These problems often undermine the results that can be obtained in practical applications of the LPV methodology leaving conversion of NL models to LPV representations to be a cumbersome procedure with many pitfalls for the regular user.[6,14]

Existing approaches for *global* LPV modeling of NL dynamical systems can be classified into two main categories: *substitution based transformation* (SBT) methods[7,15-20] and *automated conversion procedures*.[6,21-23] For a detailed comparison, see Reference 6. In general*, the existing techniques do not pay serious attention to several issues regarding the resulting LPV models, namely: how the scheduling variable and its bounds are chosen, what is the relation between these choices and the behavior of the system including the practical implementation of LPV controllers based on them, and the usefulness of the resulting LPV form for control synthesis or as a source of model structure information for identification. In addition, most techniques are based on ad-hoc mathematical manipulations (non-unique and non-systematic) and require a serious level of experience to be used.

In this article†, inspired by the strong link between feedback linearization of NL representations[12] and global LPV modeling, our objective is to provide systematic LPV embedding of the behavior of NL representations such that

- the precise relationship between the behavior of the NL representation and the LPV representation is mathematically formalized;
- the choice of p and its bounds are explicit.

Specifically, a systematic procedure is proposed to convert control affine NL-SS representations into state minimal LPV-SS representations in an observable canonical form (see Section 2). A particular advantage of this canonical form is that it can be directly converted into an equivalent LPV-IO form using the recently developed LPV realization theory[6] and hence it is highly useful for both LPV control synthesis (due to the SS form) and model structure selection in LPV identification (due to a direct LPV-IO conversion). The method is based on transforming the states of a given NL representation into a normal form such that, in the SISO case, all nonlinearities in the NL model are realized in only one NL

*Except for the decision tree algorithm in References 6 and 23.
†Preliminary ideas leading to the theorems presented in this article appeared in the conference contribution.[24]

term. Then, an exact substitution-based technique is presented to provide the LPV model (see Sections 4 and 3 for the overall procedure). The state transformation leads to the systematic construction of scheduling signals. More precisely, the scheduling signals depend either on the inputs, outputs, and their derivatives, or on some of the observable states of the original NL representation. In particular, scheduling construction based on inputs, outputs, and their derivatives compared to state-dependent scheduling is practically useful for systems (e.g., mechatronic applications) where outputs and their derivatives are directly measurable or low noise conditions enable their estimation (see Section 3.5 for details). To demonstrate the performance and limitations of the introduced conversion methods simulation and measurement examples are provided in Section 5.

2 | LPV REPRESENTATIONS

As the first step, we define the class of the considered LPV system representations and their associated solution sets, that is, behaviors, which will be used to describe/embed the solution set of nonlinear systems, further defined in Section 3.

2.1 | Mathematical preliminaries

Let $\mathcal{C}_k(\mathbb{R}, \mathbb{W})$ be the space of k-times continuously differentiable real functions $w : \mathbb{R} \rightarrow \mathbb{W} \subseteq \mathbb{R}^{n_\mathrm{w}}$ with left compact support that satisfy $\frac{d^i}{dt^i} w(t) \in \mathbb{W}$ for all $t \in \mathbb{R}$ and $i \in \mathbb{I}_1^k = \{1, \dots, k\}$. Let $\mathbb{P}$ be an open subset of $\mathbb{R}^{n_\mathrm{p}}$ and let $\mathcal{R}_k(\mathbb{P})$ denote the *set* of real-analytic functions of the form $f : \mathbb{P}^k \rightarrow \mathbb{R}$ in $n_\mathrm{p}k$ variables. For $\hat{k} > k$, any $f \in \mathcal{R}_k(\mathbb{P})$ is called equivalent with a $\hat{f} \in \mathcal{R}_{\hat{k}}(\mathbb{P})$ if $\hat{f}(\eta_1, \dots, \eta_{\hat{k}}) = f(\eta_1, \dots, \eta_k)$ for all $\eta_1, \dots, \eta_k \in \mathbb{P}$, as $\hat{f}$ is not *essentially dependent* on its arguments. Define the set operator $\ominus$, such that $\mathcal{R}_{k+1}(\mathbb{P}) \ominus \mathcal{R}_k(\mathbb{P})$ contains all $f \in \mathcal{R}_{k+1}(\mathbb{P})$ not equivalent with any element of $\mathcal{R}_k(\mathbb{P})$. This prompts to considering the set $\mathcal{R}(\mathbb{P}) = \bigcup_{k=0}^{\infty} \mathcal{R}_k(\mathbb{P}) \ominus \mathcal{R}_{k-1}(\mathbb{P})$ where $\mathcal{R}_0(\mathbb{P}) = \mathbb{R}$ and $\mathcal{R}_{-1}(\mathbb{P}) = \emptyset$. We can define addition and multiplication in $\mathcal{R}(\mathbb{P})$ analogous to that of:[25] if $f_1, f_2 \in \mathcal{R}(\mathbb{P})$, then $f_i \in \mathcal{R}_{k_i}(\mathbb{P}) \ominus \mathcal{R}_{k_i-1}(\mathbb{P})$, for some integer $k_i \geq 0$, $i = 1, 2$, and, by taking $k = \max\{k_1, k_2\}$, the equivalence described above implies that there exist equivalent representations of these functions in $\mathcal{R}_k(\mathbb{P})$. Then $f_1 + f_2, f_1 \cdot f_2$ can be defined as the usual addition and multiplication of functions in $\mathcal{R}_k(\mathbb{P})$ and the result, in terms of the equivalence, is considered to be a $f \in \mathcal{R}(\mathbb{P})$. For a $p \in \mathcal{C}_\infty(\mathbb{R}, \mathbb{P})$, we define the following notation: if $f \in \mathcal{R}(\mathbb{P})$, then $f \diamond p : \mathbb{R} \rightarrow \mathbb{R}$ is

$$\forall t \in \mathbb{R} : \quad (f \diamond p)(t) = f\left(p(t), \frac{d}{dt}p(t), \dots, \frac{d^k}{dt^k}p(t)\right), \tag{3}$$

where k is an integer such that $f \in \mathcal{R}_k(\mathbb{P}) \ominus \mathcal{R}_{k-1}(\mathbb{P})$. We denote by $\mathcal{R}^{k \times l}(\mathbb{P})$ the set of all $k \times l$ matrices whose entries are elements of $\mathcal{R}(\mathbb{P})$ which also extends the operator $\diamond$ to matrices whose entries are functions from $\mathcal{R}(\mathbb{P})$.

2.2 | State-space representation

For the sake of simplicity for defining the embedding of the dynamics of an NL system into the solution set of an LPV representation, we will introduce a slightly extended definition of LPV state-space representations compared to the regular definitions treated in the literature.[7,10]

Definition 1 (LPV-SS representation). A continuous-time LPV-SS representation with an open scheduling region $\mathbb{P}$ of dimension n_p is a tuple of matrices of analytic functions:

$$\left[\begin{array}{c|c} \mathcal{A} & \mathcal{B} \\ \hline \mathcal{C} & \mathcal{D} \end{array}\right] \in \left[\begin{array}{c|c} \mathcal{R}^{n_\mathrm{z} \times n_\mathrm{z}}(\mathbb{P}) & \mathcal{R}^{n_\mathrm{z} \times n_\mathrm{u}}(\mathbb{P}) \\ \hline \mathcal{R}^{n_\mathrm{y} \times n_\mathrm{z}}(\mathbb{P}) & \mathcal{R}^{n_\mathrm{y} \times n_\mathrm{u}}(\mathbb{P}) \end{array}\right]. \tag{4}$$

A solution of this representation is a tuple $(u, z, y, p) \in \mathcal{C}_{n_\mathrm{z}}(\mathbb{R}, \mathbb{U} \times \mathbb{Z} \times \mathbb{Y}) \times \mathcal{C}_\infty(\mathbb{R}, \mathbb{P})$ such that

$$\frac{d}{dt} z = (\mathcal{A} \diamond p) z + (\mathcal{B} \diamond p) u, \tag{5a}$$

$$y = (\mathcal{C} \diamond p) z + (\mathcal{D} \diamond p) u, \tag{5b}$$

where z is the state vector‡, $\mathbb{Z} = \mathbb{R}^{n_z}$ is the state space, $u : \mathbb{R} \to \mathbb{U} = \mathbb{R}^{n_u}$ is the input while $y : \mathbb{R} \to \mathbb{Y} = \mathbb{R}^{n_y}$ is the output of the represented system. We denote by

$$\mathfrak{B}_{SS} = \left\{(u, z, y, p) \in C_{n_z}(\mathbb{R}, \mathbb{U} \times \mathbb{Z} \times \mathbb{Y}) \times C_\infty(\mathbb{R}, \mathbb{P}) | \text{(5a)–(5b) hold} \right\}, \tag{6}$$

the solution set (latent behavior) of (5a)–(5b).

Note that in the above defined SS representation, the operator $\diamond$ expresses the dependence of the state-space matrix functions along a scheduling trajectory p and its derivatives; in other words, it expresses a dynamic mapping between p and $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$. We refer to this dynamic mapping between the scheduling signal and the system matrices as *dynamic dependence*, whereas the dependence on the value of $p(t)$ only is referred to as *static dependence*. The latter is used in the conventional definitions that can be found in the literature,[7,10] however, we need the notion of dynamic dependence here to show how systematic embedding of NL systems can be achieved by LPV models. Moreover, LPV models with dynamic dependence arise naturally as a result of system manipulations, such as state transformations, observability, controllability canonical forms, and so forth.[25] For technical reasons, in this article we work with LPV-SS representations in observable canonical form. As its name suggests, an LPV model in observable canonical form is state observable and it allows a simple conversion to *input-output* (IO) representations. The latter is important for system identification, since IO representations are easier to identify than state-space models. Conditions for existence of a state-space isomorphism transforming an LPV-SS representation to an observable canonical form are discussed in References 25,26. The matrices, associated with the observability canonical representation of (5) in the SISO case, under the assumption of minimality of (5), are given by:[6]

$$\left[\begin{array}{c|c} \mathcal{A} & \mathcal{B} \\ \hline \mathcal{C} & \mathcal{D} \end{array}\right] = \left[\begin{array}{cccc|c} 0 & 1 & \dots & 0 & \beta_{n_z-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & \beta_1 \\ \alpha_0 & \alpha_1 & \dots & \alpha_{n_z-1} & \beta_0 \\ \hline 1 & 0 & \dots & 0 & \beta_{n_z} \end{array}\right], \tag{7}$$

where $\{\alpha_i\}_{i=0}^{n_z-1}$ and $\{\beta_j\}_{j=0}^{n_z-1}$ are analytic functions in $\mathcal{R}(\mathbb{P})$. A special case of (7), when $\beta_{n_z} = \dots = \beta_1 = 0$, is given by

$$\left[\begin{array}{c|c} \mathcal{A} & \mathcal{B} \\ \hline \mathcal{C} & \mathcal{D} \end{array}\right] = \left[\begin{array}{cccc|c} 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \\ \alpha_0 & \alpha_1 & \dots & \alpha_{n_z-1} & \beta_0 \\ \hline 1 & 0 & \dots & 0 & 0 \end{array}\right], \tag{8}$$

which is of particular importance in this work as demonstrated later. In the sequel, we refer to the forms (7) and (8) as the *full and simplified observability forms*, respectively. In this article, we present a method for transforming a nonlinear system to LPV simplified observability form and another method which yields an LPV representation in full observability form.

3 | CONVERSION TO THE SIMPLIFIED OBSERVABILITY FORM

In this section, we discuss conversion of input-affine nonlinear models to simplified LPV observability canonical forms.

‡We use z to denote the state vector in an LPV-SS representation. This allows later to distinguish z from the state vector x associated with an NL-SS representation.

3.1 | The problem setting

Consider a SISO NL system $\mathcal{G}$ represented in the form of

$$\frac{d}{dt}x = f(x) + g(x)u, \tag{9a}$$

$$y = h(x), \tag{9b}$$

where $f, g : \mathbb{X} \to \mathbb{R}^{n_x}$ and $h : \mathbb{X} \to \mathbb{R}$ are real analytic functions, $\mathbb{X}$ is an open subset of $\mathbb{R}^{n_x}$ and $u : \mathbb{R} \to \mathbb{U} \subseteq \mathbb{R}$ is the input with $y : \mathbb{R} \to \mathbb{Y} \subseteq \mathbb{R}$ being the output signal and $x : \mathbb{R} \to \mathbb{X}$ is the state variable. We consider the solutions of (9) in the following sense

$$\mathfrak{C}_{\rm SS} = \left\{(u, x, y) \in C_{n_x}(\mathbb{R}, \mathbb{U} \times \mathbb{X} \times \mathbb{Y}) | \text{(9a–b) hold for all } t \in \mathbb{R}\right\}. \tag{10}$$

The form (9) represents a rather general class of NL systems, commonly referred to as input-affine systems, which includes common models of mechanical systems[27] and many first-principles models in process control.[28] More general representation of NL systems characterized by $\dot{x} = f(x, u)$, with $f : \mathbb{X} \times \mathbb{U} \to \mathbb{R}^{n_x}$ being an analytic vector field, can be rewritten in the input affine form (9) according to the procedure detailed in Reference 27. Furthermore, in (9b), there is no direct feedthrough term as, w.l.o.g., such feedthrough terms can be easily eliminated via the projection of y.

To achieve our objective, that is, to embed the dynamical behavior of NL systems represented by (9) into the solution set of an LPV-SS representation in a simplified observable canonical form given by (8), we intend to use the concept of the embedding principle discussed in Section 1 to develop multi-path feedback linearization of (9). Before going into the mathematical details, we present the main idea informally. Consider a solution (x, y, u) of (9), and define

$$z = \begin{bmatrix} y & \frac{d}{dt}y & \dots & \frac{d^{n_x}}{dt^{n_x}}y \end{bmatrix}^\top, \tag{11a}$$

$$v = \begin{bmatrix} u & \frac{d}{dt}u & \dots & \frac{d^{n_x}}{dt^{n_x}}u \end{bmatrix}^\top. \tag{11b}$$

Let (9) be observable, that is, $x = \Psi(z, v)$ for some map Ψ and let Φ (an implicit function of Ψ) be such that

$$z = \Phi(x, v). \tag{12}$$

Then, we can obtain a new state-space description of (9):

$$\frac{d}{dt}z_1 = z_2, \qquad \dots \qquad \frac{d}{dt}z_{n_x-1} = z_{n_x}, \tag{13a}$$

$$\frac{d}{dt}z_{n_x} = \lambda(z, v), \tag{13b}$$

$$y = z_1, \tag{13c}$$

where λ is an analytical function, such that if (u, x, y) is a solution of (9), then (u, z, y) is a solution of (13) with z and x related by (12). If λ can be factorized as

$$\lambda(z, v) = \beta_0(z, v)u + \sum_{i=0}^{n_x-1} \alpha_i(z, v)z_{i+1}, \tag{14}$$

for some analytic functions β_0 and $\{\alpha_i\}_{i=0}^{n_x-1}$, then by setting $p = [y \quad u]^\top$, and changing the ordering of the arguments of β_0 and $\{\alpha_i\}_{i=0}^{n_x-1}$, (13b) can be written as

$$\frac{d}{dt}z_{n_x} = \sum_{i=0}^{n_x-1} (\alpha_i \diamond p)z_{i+1} + (\beta_0 \diamond p)u, \tag{15}$$

which implies that (u, z, y, p), with z being related to x by (12) and $p = [y \quad u]^\top$, is a solution of an LPV observable canonical form (8) with $n_x = n_z$. As p of the resulting LPV-SS model is composed of the output and input signals of the system, it is measurable/available in most real-world applications, that is, the transformation yields an LPV-SS form that opens the possibility to design LPV controllers for which implementation can avoid or mitigate the need for state measurements or scheduling observers.

3.2 | Mathematical details of the construction

Below we present the ideas outlined above in a more rigorous way. First of all, note that we need to choose a point $x_0 \in \mathbb{X}$ around which the embedding can be developed and its validity can be analyzed. From the point of view of controller synthesis, it is often desirable to consider $x_0 = 0$ so that any stabilizing controller designed for the resulting LPV-SS form will aim at keeping the state of the original system in a neighborhood of x_0. To this end, we will make the following assumption.

Assumption 1 (Centering). To simplify the discussion, in the sequel, we will assume w.l.o.g. that $f(x_0) = 0$ and $h(x_0) = 0$.

Note that $f(x_0) = 0$ can easily be achieved by state and input transformation, while $h(x_0) = 0$ requires transformation of the output signal y.

Definition 2 ($(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$-admissible solutions). Let $\mathbb{X}_0$ be an open neighborhood of x_0 in $\mathbb{X}$. Furthermore, choose open sets $0 \in \mathbb{U}_0 \subseteq \mathbb{R}$, $0 \in \mathbb{Y}_0 \subseteq \mathbb{R}$. A solution $(u, x, y) \in \mathfrak{C}_{SS}$ of (9) is said to be $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$-admissible, if $u \in C_{n_x}(\mathbb{R}, \mathbb{U}_0)$, $x \in C_\infty(\mathbb{R}, \mathbb{X}_0)$ and $y \in C_{n_x}(\mathbb{R}, \mathbb{Y}_0)$.

Next, we recall from References 29,30 the notion of local uniform observability.

Definition 3 (Local uniform observability). The representation (9) is called locally uniformly observable on the open sets $x_0 \in \mathbb{X}_0 \subseteq \mathbb{R}^{n_x}$, $0 \in \mathbb{U}_0 \subseteq \mathbb{R}$, $0 \in \mathbb{Y}_0 \subseteq \mathbb{R}$, if there exists an analytic map

$$\Psi : (\mathbb{Y}_0 \times \mathbb{U}_0)^{n_x} \to \mathbb{X}_0, \tag{16}$$

such that for any $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$-admissible solution (u, x, y) of (9), it holds that

$$x = \Psi\left(\begin{bmatrix} y \\ u \end{bmatrix}, \frac{d}{dt}\begin{bmatrix} y \\ u \end{bmatrix}, \ldots, \frac{d^{n_x-1}}{dt^{n_x-1}}\begin{bmatrix} y \\ u \end{bmatrix}\right). \tag{17}$$

We will call the map Ψ the $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$-*observability map* or *observability map*, if $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$ is clear from the context and call (9) *locally uniformly observable*, if it is locally uniformly observable on $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$ for some open sets $\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0$.

If (9) is locally uniformly observable, then it is possible to express the n_xth derivative of its output y as a function of $\left\{\frac{d^i}{dt^i}y\right\}_{i=0}^{n_x-1}$ and $\left\{\frac{d^j}{dt^j}u\right\}_{j=0}^{n_x-1}$. In order to present the construction formally, we define the following collection of functions.

Definition 4 (Output derivative function). For each $k \in \mathbb{N}$, define the functions $\Phi_k : \mathbb{X} \times \mathbb{U}^k \to \mathbb{Y}$ as follows:

$$\Phi_0(x) = h(x), \tag{18a}$$

$$\Phi_k(x, v_1, \ldots, v_k) = \sum_{i=1}^{n_x} \left[(f_i(x) + g_i(x)v_1) \frac{\partial \Phi_{k-1}}{\partial x_i}(x, v_1, \ldots, v_{k-1}) + \sum_{j=1}^{k-1} v_{j+1} \frac{\partial \Phi_{k-1}}{\partial v_j}(x, v_1, \ldots, v_{k-1}) \right], \tag{18b}$$

where f_i and g_i denote the ith element of these functions. The map Φ_k will be called the kth output derivative map.

For any $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$-admissible solution (u, x, y) of (9):

$$\frac{d^k}{dt^k}y = \Phi_k\left(x, u, \frac{d}{dt}u, \ldots, \frac{d^{k-1}}{dt^{k-1}}u\right), \tag{19}$$

which leads to the following corollary:

Corollary 1 (NL-IO realization). *If (9) is locally uniformly observable on* $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$ *with the observability function* Ψ, *then for any* $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$*-admissible solution* (u, x, y) *of (9):*

$$\frac{d^{n_x}}{dt^{n_x}}y = \Gamma_{n_x}\left(\begin{bmatrix} y \\ u \end{bmatrix}, \frac{d}{dt}\begin{bmatrix} y \\ u \end{bmatrix}, \dots, \frac{d^{n_x-1}}{dt^{n_x-1}}\begin{bmatrix} y \\ u \end{bmatrix}\right), \tag{20}$$

where the analytic map $\Gamma_{n_x} : (\mathbb{Y}_0 \times \mathbb{U}_0)^{n_x} \to \mathbb{Y}_0$ *is defined by*

$$\Gamma_{n_x}\left(\begin{bmatrix} \eta_1 \\ v_1 \end{bmatrix}, \dots, \begin{bmatrix} \eta_{n_x} \\ v_{n_x} \end{bmatrix}\right) = \Phi_{n_x}\left(\Psi\left(\begin{bmatrix} \eta_1 \\ v_1 \end{bmatrix}, \dots, \begin{bmatrix} \eta_{n_x} \\ v_{n_x} \end{bmatrix}\right), v_1, \dots, v_{n_x}\right), \tag{21}$$

for all $\eta_1, \dots, \eta_{n_x} \in \mathbb{Y}_0$ *and* $v_1, \dots, v_{n_x} \in \mathbb{U}_0$.

Corollary 1 paves the way to represent $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$-admissible solutions of (9) as solutions of an LPV observer canonical form. In order to present the precise result, we have to introduce some concepts related to factorization of functions.

Note that for a given open set $\mathbb{V} \subseteq \mathbb{R}^n$, any analytic function $f : \mathbb{V} \to \mathbb{R}$ can be decomposed as

$$f(\xi) = \frac{N(\xi, \phi_1(\xi), \dots, \phi_\tau(\xi))}{D(\xi, \phi_1(\xi), \dots, \phi_\tau(\xi))}, \quad \forall \xi \in \mathbb{V}, \tag{22}$$

where ξ is the indeterminate of f, N, and D are polynomial maps: $\mathbb{R}^{n+\tau} \to \mathbb{R}$ and $\{\phi_i : \mathbb{V} \to \mathbb{R}\}_{i=1}^{\tau}$ are analytic functions. If (22) holds, we will say that f *is rational w.r.t.* $\{\phi_i\}_{i=1}^{\tau}$. Note that if the functions $\{\phi_i\}_{i=1}^{\tau}$ are algebraically independent and f is rational w.r.t. to $\{\phi_i\}_{i=1}^{\tau}$, then there is a unique pair of co-prime polynomials (N, D) which satisfies (22).

Definition 5 (Factorization). Consider a given open set $\mathbb{V} \subseteq \mathbb{R}^n$ and an analytic function $f : \mathbb{V} \to \mathbb{R}$, rational w.r.t. some analytic $\{\phi_i\}_{i=1}^{\tau}$ in terms of (22). Under $\{\phi_i\}_{i=1}^{\tau}$, factorization of f with respect to the first m variables is a tuple $(\{r_i : \mathbb{V} \to \mathbb{R}\}_{i=1}^{m}, s : \mathbb{V} \to \mathbb{R})$ of analytic functions such that $r_i = M_i/D$ and $s = S/D$ in terms of (22) with $\{M_i\}_{i=1}^{m}$, D and S being polynomials in $n + \tau$ variables $X_1, \dots, X_{n+\tau}$ such that

$$N = M_1X_1 + \cdots + M_mX_m + S, \tag{23}$$

and, for all $i \in \mathbb{I}_1^m$, M_i does not depend on $\{X_l\}_{l=i+1}^{m}$ and S does not depend on $\{X_l\}_{l=1}^{m}$.

The polynomials $\{M_i\}_{i=1}^{m}$ are the result of the division of N by $\{X_l\}_{l=1}^{m}$ and S is the remainder of this division, in the sense of Reference 31(theorem 3, pp. 61-62). As $\{X_l\}_{l=1}^{m}$ are monomials, a simplified form of the algorithm described in Reference 31 is available to compute the factorization (see Algorithm 1 later). Note that if f is rational with respect to $\{\phi_i\}_{i=1}^{\tau}$, then a factorization $(\{r_i\}_{i=1}^{m}, s)$ with respect to the first m variables always exists in the form of $f(\xi) = \sum_{i=1}^{m} r_i(\xi)\xi_i + s(\xi)$. This factorization depends on $\{\phi_i\}_{i=1}^{\tau}$, that is, different choices of these functions will lead to different factorizations, the consequences of which will be discussed in Section 3.3.

Introduce the selection matrix[§] $\mathcal{R} \in \mathbb{R}^{2n_x \times 2n_x}$, which rearranges the arguments of $\Gamma_{n_x}(\zeta) : (\mathbb{Y}_0 \times \mathbb{U}_0)^{n_x} \to \mathbb{Y}_0$ such that $\Gamma_{n_x}(\mathcal{R}\xi) : \mathbb{Y}_0^{n_x} \times \mathbb{U}_0^{n_x} \to \mathbb{R}$ is equivalent with Γ_{n_x}. Formally this means that for $\eta_1, \dots, \eta_{n_x} \in \mathbb{Y}_0$ and $v_1, \dots, v_{n_x} \in \mathbb{U}_0$, $\zeta = \begin{bmatrix} \eta_1 & v_1 & \dots & \eta_{n_x} & v_{n_x} \end{bmatrix} = \mathcal{R}\xi$ where $\xi = \begin{bmatrix} \eta_1 & \dots & \eta_{n_x} & v_1 & \dots & v_{n_x} \end{bmatrix}$. We identify the resulting function as $\Gamma_{n_x} \circ \mathcal{R}$. Furthermore, consider a set of functions $\{f_i : \mathbb{W}^l \to \mathbb{R}\}_{i=1}^{\tau}$, where $\mathbb{W} \subseteq \mathbb{R}^n$ is not necessarily open. The matrix $\mathcal{T} \in \mathbb{R}^{m \times n}$, $m \leq n$, is called the *selection matrix of the essential support of* $\{f_i\}_{i=1}^{\tau}$ *under* $\mathbb{W}$, if $\mathcal{T}$ has full row rank, and the functions $\{f_i(\zeta_1, \dots, \zeta_l)\}_{i=1}^{\tau}$ with $\zeta_j \in \mathbb{W}$ depend only[¶] on $\mathcal{T}\zeta_j$. For example, if $f : \mathbb{R}^4 \to \mathbb{R}$ depends only on its first and third arguments, then $\mathcal{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ is a selection matrix of the essential support of f under $\mathbb{R}^4$, while $\mathcal{T} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ is the selection matrix under $\mathbb{R}^2$. If $\mathcal{T}$ is a selection matrix for the essential support for $\{f_i\}_{i=1}^{\tau}$, then $\mathcal{T}^{-1} = \mathcal{T}^\top$ is a selection matrix such that $\mathcal{T} \cdot \mathcal{T}^{-1} = I$ and we can identify the functions $\{f_i\}_{i=1}^{\tau}$ with the functions $\{f_i \circ \mathcal{T}^{-1}\}_{i=1}^{\tau}$. Note that while the former are functions of $n \cdot l$ variables, the latter have $m \cdot l \leq n \cdot l$ variables.

[§]A selection matrix contains zeros and a single element 1 in each row.

[¶]$\forall \{\zeta_j^{(1)}, \zeta_j^{(2)} \in \mathbb{W}\}_{j=1}^{l}$ and $\forall i \in \mathbb{I}_1^{\tau}$, $\zeta_j^{(1)} - \zeta_j^{(2)} \in \ker \mathcal{T}$ for all $j \in \mathbb{I}_1^{l} \Rightarrow f_i(\zeta_1^{(1)}, \dots, \zeta_l^{(1)}) = f_i(\zeta_1^{(2)}, \dots, \zeta_l^{(2)})$ for all $i \in \mathbb{I}_1^{\tau}$

Theorem 1 (LPV embedding, simp. observability form). *Assume that (9) is locally uniformly observable on* $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$ *with observability function* Ψ. *Furthermore, assume that there exists a set of analytic functions* $\{\phi_i : \mathbb{Y}_0^{n_x} \times \mathbb{U}_0^{n_x} \to \mathbb{R}\}_{i=1}^{\tau}$ *such that the map* $\Gamma_{n_x} \circ \mathcal{R}$ *in (20) is rational with respect to* $\{\phi_i\}_{i=1}^{\tau}$. *Let* $\left(\{r_i\}_{i=1}^{n_x+1}, s\right)$ *be a factorization of* $\Gamma_{n_x} \circ \mathcal{R}$ *with respect to the first* $n_x + 1$ *variables. If* $s = 0$, *that is, factorization is possible without a remainder and* $\mathcal{T}$ *is the essential support of* $\{r_i \circ \mathcal{R}^{-1}\}_{i=1}^{n_x+1}$ *under* $\mathbb{Y}_0 \times \mathbb{U}_0$, *then the LPV-SS representation (8) with*

$$p = \mathcal{T}[y^\top \quad u^\top]^\top, \tag{24a}$$

$$\{\alpha_i := r_{i+1} \circ \mathcal{R}^{-1} \circ \mathcal{T}^{-1}\}_{i=0}^{n_x-1}, \quad \beta_0 := r_{n_x+1} \circ \mathcal{R}^{-1} \circ \mathcal{T}^{-1}, \tag{24b}$$

and scheduling region $\mathbb{P} = \mathcal{T}(\mathbb{Y}_0 \times \mathbb{U}_0)$ *satisfies*

$$\mathfrak{C}_{\mathrm{SS}}^{\mathrm{o}} \subseteq \pi_p \mathfrak{B}_{\mathrm{SS}}^{\mathrm{o}}, \tag{24c}$$

where

$$\pi_p \mathfrak{B}_{\mathrm{SS}}^{\mathrm{o}} = \Big\{ (u, x, y) \in \mathcal{C}_{n_x}(\mathbb{R}, \mathbb{U}_0 \times \mathbb{X}_0 \times \mathbb{Y}_0) | \exists p \in \mathcal{C}_{n_x}(\mathbb{R}, \mathbb{P}), \exists z \in \mathcal{C}_{n_x}(\mathbb{R}, \mathbb{Y}_0^{n_x}) \text{ such that (5a–b) hold while } x = \Psi\left(z, u, \dots, \frac{d^{n_x}}{dt^{n_x}} u\right) \Big\},$$

and

$$\mathfrak{C}_{\mathrm{SS}}^{\mathrm{o}} = \Big\{ (u, x, y) \in \mathcal{C}_{n_x}(\mathbb{R}, \mathbb{U}_0 \times \mathbb{X}_0 \times \mathbb{Y}_0) \text{ such that (9a–b) hold} \Big\}.$$

In terms of Theorem 1, the set of all $(\mathbb{U}_0 \times \mathbb{X}_0 \times \mathbb{Y}_0)$ admissible solutions of (9) can be embedded into the solution set of an LPV-SS representation and (24a) gives a direct selection of the scheduling variables under the factorization w.r.t. $\{\phi_i\}_{i=1}^{\tau}$.

Proof. Consider a $(\mathbb{U}_0 \times \mathbb{X}_0 \times \mathbb{Y}_0)$ admissible solution (u, x, y) of (9) and invoke the definitions (11). Let $\xi = [z^\top \ v^\top]^\top$ and $\zeta = \left[\begin{bmatrix} y \\ u \end{bmatrix}, \frac{d}{dt} \begin{bmatrix} y \\ u \end{bmatrix}, \dots, \frac{d^{n_x-1}}{dt^{n_x-1}} \begin{bmatrix} y \\ u \end{bmatrix} \right]$. Notice that $\zeta = \mathcal{R}\xi$ and $\xi = \mathcal{R}^{-1}\zeta$. Introduce $\mathcal{P}$ and $\mathcal{P}^{-1}$ which are n_x-times block diagonal matrices of $\mathcal{T}$ and $\mathcal{T}^{-1}$, respectively. Notice that $\mathcal{P}\mathcal{P}^{-1}\mathcal{P}\mathcal{R}\xi = \mathcal{P}\mathcal{R}\xi$ and hence $\mathcal{P}(\mathcal{R}\xi - \mathcal{P}^{-1}\mathcal{P}\mathcal{R}\xi) = 0$. From the definition of the selection matrices it follows that

$$r_i \circ \mathcal{R}^{-1}(\mathcal{R}\xi) = r_i \circ \mathcal{R}^{-1}(\mathcal{P}^{-1}\mathcal{P}\mathcal{R}\xi) = r_i \circ \mathcal{R}^{-1} \circ \mathcal{T}^{-1}(\mathcal{P}\mathcal{R}\xi).$$

Define $\tilde{p} = [y^\top \ u^\top]^\top$. Notice that

$$\mathcal{P}\mathcal{R}\xi = \mathcal{P}\zeta = \left[(\mathcal{T}\tilde{p})^\top \quad \dots \quad \tfrac{d^{n_x-1}}{dt^{n_x-1}} (\mathcal{T}\tilde{p})^\top \right]^\top = \left[p^\top \quad \dots \quad \tfrac{d^{n_x-1}}{dt^{n_x-1}} p^\top \right]^\top.$$

Hence,

$$r_i(\xi) = r_i \circ \mathcal{R}^{-1} \circ \mathcal{T}^{-1}(\mathcal{P}\mathcal{R}\xi) = \begin{cases} \alpha_{i-1} \diamond p, & i \in \mathbb{I}_1^{n_x}; \\ \beta_0 \diamond p, & i = n_x + 1. \end{cases}$$

From the discussion above and using $\frac{d}{dt} z_i = z_{i+1}$ for $i = \mathbb{I}_1^{n_x-1}$ it follows that

$$\frac{d^{n_x}}{dt^{n_x}} z_{n_x} = \Gamma_{n_x}\left(\begin{bmatrix} y \\ u \end{bmatrix}, \dots, \frac{d^{n_x-1}}{dt^{n_x-1}} \begin{bmatrix} y \\ u \end{bmatrix} \right) = \sum_{i=1}^{n_x} r_i(\xi) z_i + r_{n_x+1}(\xi) u = \sum_{i=0}^{n_x-1} (\alpha_i \diamond p) z_{i+1} + (\beta_0 \diamond p) u. \tag{25}$$

Hence, (u, z, y, p) is a solution of the LPV-SS representation (8) defined in the statement of the theorem. Moreover, since Ψ is a $(\mathbb{U}_0 \times \mathbb{X}_0 \times \mathbb{Y}_0)$ observability function and (25) holds, $x = \Psi\left(z, u, \dots, \frac{d^{n_x-1}}{dt^{n_x-1}} u\right)$. ■

Algorithm 1. Factorization

Require: $N(X_1, \dots, X_{n+\tau}), D(X_1, \dots, X_{n+\tau}), \{\phi_i\}_{i=1}^{\tau}, m \leq n$
$S \leftarrow N$.
for $k \leftarrow m : 1$ **do**
represent S as $\sum_{(i_1,\dots,i_{n+\tau})\in\mathbb{I}} \gamma_{i_1,\dots,i_{n+\tau}} X_1^{i_1} \cdots X_{n+\tau}^{i_{n+\tau}}$ for a finite index set $\mathbb{I} \subseteq \mathbb{N}^{n+\tau}$.
$M_k \leftarrow \sum_{(i_1,\dots,i_{n+\tau})\in\mathbb{I}, i_k \geq 1} \gamma_{i_1,\dots,i_{n+\tau}} \frac{X_1^{i_1}\dots X_{n+\tau}^{i_{n+\tau}}}{X_k}$.
$S \leftarrow S - M_k X_k$.
end for
$r_i(\xi) \leftarrow \frac{M_i(\xi,\phi_1(\xi),\dots,\phi_\tau(\xi))}{D(\xi,\phi_1(\xi),\dots,\phi_\tau(\xi))}$, $s(\xi) \leftarrow \frac{S(\xi,\phi_1(\xi),\dots,\phi_\tau(\xi))}{D(\xi,\phi_1(\xi),\dots,\phi_\tau(\xi))}$, $\xi \in \mathbb{V}$.
return $\left(\{r_i\}_{i=1}^{m}, s\right)$.

In order to make Theorem 1 applicable, we need an algorithm to compute the factorization of the function $\Gamma_{n_x} \circ \mathcal{R}$ on $\mathbb{V} = \mathbb{Y}_0^{n_x} \times \mathbb{U}_0^{n_x}$ with respect to $\{\phi_i : \mathbb{V} \to \mathbb{R}\}_{i=1}^{\tau}$. Let N and D be such polynomials that $\Gamma_{n_x} \circ \mathcal{R}$ can be written as (22). Then, Algorithm 1, which takes N and D and $\{\phi_i\}_{i=1}^{\tau}$ as parameters, returns a factorization $(\{r_i\}_{i=1}^{m}, s)$ of $\Gamma_{n_x} \circ \mathcal{R}$ with respect to the first $m = n_x + 1$ variables, that is, $\left\{z_i = \frac{d^i}{dt^i} y\right\}_{i=1}^{n_x}$ and u.

Theorem 1 indicates that it is possible to embed NL systems into LPV-SS representations in a systematic way. Furthermore, it characterizes an LPV embedding in terms of a multi-path linearization which resembles feedback linearization of NL systems. However, in feedback linearization, a virtual input signal is introduced so that the transformed system becomes LTI. In contrast, in the proposed LPV approach, a set of virtual variables, denoted by p, are constructed which result in a varying linear relationship. Thus, the obtained LPV-SS representation is useful to develop controllers that can shape the closed-loop behavior unrestricted or have better robustness than with an LTI target behavior. Furthermore, p is selected to be state-independent (in contrast with the common NL to LPV conversion techniques) meaning that in practice, the LPV controller designed for this model can be potentially directly applied in a real-world system without the need of a scheduling observer (see Section 3.5 for a detailed discussion). Furthermore, the dimension of p is reduced by considering the essential support of $\{r_i\}_{i=1}^{n_x+1}$. On the other hand, Theorem 1 guarantees the embedding and hence the validity of the LPV representation only for those state trajectories x of the NL system which remain in $\mathbb{X}_0$ and for those inputs u which remain in $\mathbb{U}_0$. Hence, when designing controllers using the LPV-SS form, one must ensure that $u(t) \in \mathbb{U}_0$ and x remains in $\mathbb{X}_0$. For the latter, it is enough to ensure that the state z of the LPV-SS model remains in $\mathbb{Y}_0^{n_x}$. Otherwise, the LPV-SS representation of the NL system is no longer valid.

3.3 | Choice of the scheduling variable

Although Theorem 1 gives a straightforward formulation of the LPV-SS representation of (9) with a unique choice of p, one may consider projections of this variable to simplify the resulting dependency structure of (9) as follows:

- *Full dynamic dependency*: (24) results in a possible dynamic dependence of (4) on $p = \mathcal{T}\begin{bmatrix} y & u \end{bmatrix}^\top$ with $\mathbb{P} = \mathcal{T}(\mathbb{Y}_0 \times \mathbb{U}_0) \subseteq \mathbb{R}^m$, $m \leq n_y + n_u$, characterized by rational combinations of the chosen $\{\phi_i\}_{i=1}^{\tau}$. Although such a choice is tempting from the theoretical and even identification point of view, as it minimizes the conservativeness of the embedding, it results in models which are difficult for control design. Current techniques are only able to handle rational static dependence on p.
- *Rational dependence*: Using the "minimal" scheduling choice characterized by Theorem 1, it is possible to introduce a so-called *scheduling map* μ:

$$p = \mu \diamond (y, u) = \left[\mathcal{T}\begin{bmatrix} y & u \end{bmatrix}^\top \quad \phi_1\left(\mathcal{T}\begin{bmatrix} y & u \end{bmatrix}^\top, \dots, \frac{d^{n_x-1}}{dt^{n_x-1}}\mathcal{T}\begin{bmatrix} y & u \end{bmatrix}^\top\right) \quad \dots \quad \phi_\tau\left(\mathcal{T}\begin{bmatrix} y & u \end{bmatrix}^\top, \dots, \frac{d^{n_x-1}}{dt^{n_x-1}}\mathcal{T}\begin{bmatrix} y & u \end{bmatrix}^\top\right)\right]^\top. \quad (26)$$

Hence, by increasing $\dim(p)$ to $m + \tau$, where m is the number of rows in $\mathcal{T}$, the dynamic nature of the dependence can be hidden into μ and the p-dependence of (4) is reduced to be static rational. This is desirable for control and identification as μ can be applied on the measured values of (u, y) to compute p. Note that increasing the dimensions of p leads to more conservatism as $\pi_p \mathfrak{B}_{SS}^{o}$ grows with every hidden relation in μ.

- *Affine dependence*: The previous procedure can also be applied to hide even the polynomial dependence resulting from the above mentioned procedure by constructing a map $p = \mu \diamond (y, u)$ which, by substituting it to (24b), results in an affine dependence of (4) on p. While this is tempting to simplify control synthesis based on such an embedding, it also maximizes the conservativeness of $\pi_p \mathfrak{B}_{SS}^{o}$.

Note that computation of the analytic map Γ_{n_x} requires inversion of functions, and hence in general, it is not guaranteed that it has a closed form. While theoretically this does not hinder the application of Theorem 1, it makes the calculation of the LPV model described in Theorem 1 far from trivial. In principle, what is required for Theorem 1 is not an analytic expression for Γ_{n_x}, but an expression for the factorization of Γ_{n_x}. The latter might be computable even if there is no analytic expression for Γ_{n_x}.

In conclusion, Theorem 1 reveals that LPV embedding of an NL system is affected by a trade-off between conservativeness and the simplicity of dependence of the resulting representation on p. In this respect, it is interesting to observe that the choice of basis functions $\{\phi_i\}_{i=1}^{\tau}$ does not influence the validity of the transformation nor the controllability or observability of the resulting model as long as there is no remainder term, that is, $s = 0$. However, when $\{\phi_i\}_{i=1}^{\tau}$ are absorbed into μ, their choice has a significant impact on the conservativeness of the embedding. As in system identification, the choice of μ is invisible for the estimation procedure and it can seriously affect the outcome of the estimation (persistency of excitation, correlation with noise, etc.), while in control, robustness of the control law can be analyzed against variations of the LPV-SS representation, but not against variations in μ. Additionally, in LPV-MPC, hidden relations in μ, especially dependence on u, can seriously compromise the meaningfulness of the resulting optimization problem; hence, in principle, control design and LPV model development, in terms of the choice of μ should be seen as a joint process, see References 23,32.

3.4 | Handling the remainder term

Theorem 1 deals with the case when $s = 0$, that is, Γ_{n_x} can be factorized without a remainder. Suppose that the conditions of Theorem 1 hold, but $s \neq 0$. In this case, we can still represent the solutions of (9) by solutions of an LPV system (similarly to Theorem 1), but the resulting representation will not be linear due to the extra p-dependent affine term $\gamma := s \circ \mathcal{R}^{-1} \circ \mathcal{T}^{-1}$. This term is undesirable both in LPV control synthesis and identification as the whole LPV framework builds upon the assumed linearity of the system description. As this phenomenon is not uncommon in applied LPV control, we collected here the possible strategies to deal with affine terms:

- *Virtual input*: An input-disturbance signal $d \equiv 1$ is introduced to incorporate the affine term into the $\mathcal{B}$ matrix:

$$\tilde{\mathcal{B}} \diamond p = \begin{bmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ (\beta_0 \diamond p) & (\gamma \diamond p) \end{bmatrix} \quad \text{with new input:} \quad \begin{bmatrix} u \\ d \end{bmatrix}.$$

  Then, considering d as a time-varying disturbance with an $\mathcal{L}_2$ norm bound of 1, optimal control synthesis or MPC control can be conveniently applied. Although this strategy changes the IO partition of the system and it increases the conservativeness of the embedding, it leads to a complete representation of the original NL behavior.
- *Ignored* in the LPV “representation” of the system behavior and during control synthesis one of the following choices are applied
  - The designed controller is augmented with a feedforward path to compensate for γ during control implementation, see References 33,34.
  - Input disturbance rejection is considered as a control objective.
- *Enforced factorization*: γ is rewritten as $\frac{\tilde{\gamma}}{u} u$ or $\frac{\tilde{\gamma}}{z_j} z_j$ and added to β_0 or α_j, respectively. The associated u or z_j should never approach close to the origin during operation, otherwise loss of stability might occur, see References 6,22 for more details.

3.5 | Implementation of the scheduling

The introduced multi-path feedback realization has resulted in a systematic LPV conversion method where in terms of (25), computation of the p-dependence of the resulting LPV-SS representation (4) (irrespective how p is extracted via μ) can potentially need $n_x - 1$ time derivatives of (y, u). One can argue that computation of such derivatives based on noisy measurements can be difficult in practice. However, as we intend to show, such construction of p in fact opens up novel implementation possibilities of LPV control and identification in practice, and in principle it is not worse than scheduling constructions relaying on the state.

Control design: If an LPV controller $\mathcal{K}$ is synthesized for the LPV model resulting from the proposed conversion scheme, then through its dependence on p, implementation of $\mathcal{K}$ will require the computation of p dependent on the derivatives of (y, u). Derivatives of u correspond to derivatives of the output of $\mathcal{K}$, which can be obtained by an extended state realization of $\mathcal{K}$. Regarding derivatives of y, the following options are available:

- *Direct measurement:* In many applications, low order derivatives of the output are directly measurable. For mechatronic systems, the underlying kinematic and electric IO relationships are 2nd-order in nature and often measurements of the involved variables such as velocity and acceleration are available (e.g., via IMUs, various designs of gyroscopic, piezoelectric, optical, magnetic, radar, and ultrasonic sensors). Rate measurements are also not uncommon in many thermal, hydraulic, chemical, and biological systems especially for flow variables and, by changing the state basis, an equivalent representation can be found where the states can qualify as derivatives of the actual measured output of such systems.
- *Numerical differentiation and filtering methods*: In case the required derivatives of y are not directly measurable, numerical differentiation can be applied together with filtering methods to mitigate the effect of noise and approximation error on the computation of the derivatives (see e.g., References 35-40).
- *Observer design:* The NL model of the plant dynamics can be transformed to an observability form where the state variables directly correspond to the derivatives of y up to the relative degree of the system and the rest of the state variables can be used to compute higher derivatives of y when the derivatives of u are known. This means that derivatives of y can be estimated by an observer or a Kalman filter as any other state variables. Commonly derivatives of y naturally appear among the state variables of first-principles based plant models, like position, velocity, acceleration in motion equations of mechanical systems.

Identification: When identification of the resulting LPV model is considered in continuous time, computation of time-derivatives of (y, u) in either frequency domain or time-domain, in prediction or simulation, are required by most identification methods (subspace methods, prediction-error minimization, instrumental variables, etc.). Therefore, handling derivatives of (y, u) is a natural step in many cases, only the means of obtaining them differs which ranges from numerical differentiation and filtering to multiplying the frequency spectrum with $\mathrm{i}\omega$. Note that for LPV system identification, non-state-dependent constructions of p are advantageous in general, because due to the absence of the system model, whose estimation is the objective of the identification approach, computation of p w.r.t. a non-directly measured state variables is not possible.

Effect of measurement noise: Note that effect of measurement noise of (y, u) influences the computation of p whether the elements of p are directly measured, obtained via numerical differentiation and filtering or estimated via an observer. The resulting noise or reconstruction error on p is highly dependent on the actual system and applied sensors hence the resulting tradeoffs between the listed computation schemes are application specific. For example, in case of high-resolution encoders, computation of high-order output derivatives via numerical differentiation is feasible, while for chemical systems, direct measurement of composition has relatively high noise and requires sensor fusion and appropriate filtering to be used as a scheduling. Hence for the latter case, observer based estimation of p is often required. Analyzing that which computational approach to be applied for reconstruction of p and how the resulting error influences the outcome of applied LPV control and identification based on the proposed conversion approach is beyond the scope of the current article. Measurement noise or reconstruction error of p influences application of all LPV control and identification methods in general, irrespective of how p is chosen. Despite of an increasing research effort (see e.g., References 41-43), there is no comprehensive performance analysis framework available for general nonlinear systems regarding these effects.

Comparison to existing methods: Alternative conversion methods to LPV form often choose state-variables of the NL model in an ad-hoc manner to be part of p. With such a choice, p is often not measurable and the LPV controller $\mathcal{K}$

has to be used together with an observer for estimating p. However, $\mathcal{K}$ is designed with the assumption that p is known. Hence, by introducing an observer for estimating p, the stability and performance guarantees of the LPV controller are lost and nonlinear analysis is required to regain them. Compared to the case when p is constructed from (y, u) and their measurable derivatives, such as with the proposed method, this problem is avoided. In case specific components of p are computed with numerical differentiation, the introduced numerical noise on p can be well characterized in view of the measurement noise and can be directly taken into account of robustness analysis of the resulting closed-loop behavior. In case of relatively high noise and high-order derivatives of y, the proposed construction of p loses its advantage over state-dependent scheduling construction. However, in such cases, one can design an observer for an observability form of the plant dynamics by which direct estimation of high-order derivatives of y becomes available. As for any NL state-representation, the state can be only successfully tracked in $\mathbb{X}_0$ if the representation is locally-observable in $(\mathbb{U}_0, \mathbb{X}_0, \mathbb{Y}_0)$, hence the observability form is isomorph with the original system model in this region and there is no theoretical advantage in designing an observer for the original state or the state of the observability form which corresponds to the derivatives of y. Of course, one can argue that delay and performance loss can also be introduced with numerical differentiation and filtering methods or with the observer design in case derivatives of y are not directly measurable. However, such delays and performance losses also occur in case of tracking state-dependent scheduling variables. When compared to nonlinear control, the same choice occurs in feedback linearization when one can choose between using derivatives of y or the states x of the original system representation to calculate the linearizing feedback. The proposed methodology in this article aims at providing systematic options beyond using only x in the scheduling map and provides more attractive scheduling construction for systems where low order derivatives of y are directly measurable or can be accurately estimated.

3.6 | Computation of Γ_{n_x}

For the sake of completeness, the construction procedure of Γ_{n_x}, which is used in Theorem 1 and relies on known NL system theory concepts, is presented next.

Definition 6 (Relative degree[12]). The NL-SS system representation (9) is said to have relative degree n_r at a point $x_0 \in \mathbb{X}$ if there exists an open subset $x_0 \in \mathbb{X}_r \subseteq \mathbb{X}$ such that

(i) $L_g L_f^i h(x) = 0,\ \forall x \in \mathbb{X}_r,\ i < n_r - 1,$
(ii) $L_g L_f^{n_r-1} h(x_0) \neq 0,$

where $L_f^i h(\cdot)$ stands for the ith Lie-derivative of h w.r.t. f .

Note that not every NL system represented in the form of (9) has a relative degree n_r at all. Neither is it true that the same n_r qualifies for all $x_0 \in \mathbb{X}$. We refer to Reference 12 for more in depth discussion on this topic. In the sequel, it is assumed that x_0 is chosen such that the relative degree of (9) is well-defined at this point. Next, we consider the construction of Γ_{n_x} in a neighborhood of x_0 in two cases: when n_r of (9) at x_0 equals n_x and when $n_r < n_x$.

3.6.1 | Case of $n_r = n_x$

Consider a solution (u, x, y) of (9), such that for all $t \in \mathbb{R}$, $x(t) \in \mathbb{X}_r$ (see Definition 6). In this case,

$$z_1 = y = h(x) = \Phi_0(x), \tag{27a}$$

$$\vdots$$

$$z_{n_x} = \frac{d^{n_x-1}}{dt^{n_x-1}} y = L_f^{n_x-1} h(x) = \Phi_{n_x-1}(x), \tag{27b}$$

while

$$\frac{d^{n_x}}{dt^{n_x}} y = L_f^{n_x} h(x) + L_g L_f^{n_x-1} h(x) u = \Phi_{n_x}(x, u), \tag{27c}$$

that is, only the n_x^{th} derivative of y depends on u. This gives

$$z = \Phi(x) = \begin{bmatrix} h(x) & L_f h(x) & \dots & L_f^{n_x-1} h(x) \end{bmatrix}^\top, \tag{28}$$

hence the local inverse of Φ provides the observability function Ψ in Definition 3 to construct Γ_{n_x}. Recall[12(lemma 4.1.1,p. 140)] that if the relative degree n_r of (9) is n_x at x_0, then the gradients $\nabla h(x_0), \dots, \nabla L_f^{n_x-1} h(x_0)$ are linearly independent. Hence, in this case, the Jacobian of $\Phi(x_0)$ is invertible based on the *inverse function theorem*:[44]

Lemma 1 (Inversion of Φ). *There exist open sets $\mathbb{X}_0 \subseteq \mathbb{X}_r$ and $\mathbb{Y}_0 \subseteq \mathbb{R}$, such that $x_0 \in \mathbb{X}_0$, $\Phi(\mathbb{X}_0) = \mathbb{Y}_0^{n_x}$ and $\Phi|_{\mathbb{X}_0} : \mathbb{X}_0 \to \mathbb{Y}_0^{n_x}$, the restriction of Φ to $\mathbb{X}_0$, is an analytic diffeomorphism, that is, the analytic inverse $\Phi^\dagger$ of $\Phi|_{\mathbb{X}_0}$ exists.*

By a slight abuse of notation, we will identify Φ in the sequel with its restriction to $\mathbb{X}_0$, that is, we will view it as a diffeomorphism $\Phi|_{\mathbb{X}_0} : \mathbb{X}_0 \to \mathbb{Y}_0^{n_x}$. Let $\mathbb{U}_0$ be an arbitrary open subset of $\mathbb{U}$. We can define the observability map $\Psi : (\mathbb{Y}_0 \times \mathbb{U}_0)^{n_x} \to \mathbb{X}_0$, satisfying Definition 3, by

$$\Psi\left(\begin{bmatrix} \eta_1 \\ \upsilon_1 \end{bmatrix}, \dots \begin{bmatrix} \eta_{n_x} \\ \upsilon_{n_x} \end{bmatrix}\right) = \Phi^\dagger\left(\eta_1, \dots, \eta_{n_x}\right), \tag{29}$$

for all $\eta_1, \dots, \eta_{n_x} \in \mathbb{Y}_0$ and $\upsilon_1, \dots, \upsilon_{n_x} \in \mathbb{U}_0$. Note that, in this case, Ψ does not depend on $\{\upsilon_i\}_{i=1}^{n_x}$. Hence, by the construction in Theorem 1, Γ_{n_x} results in

$$\Gamma_{n_x}\left(\begin{bmatrix} \eta_1 \\ \upsilon_1 \end{bmatrix}, \dots, \begin{bmatrix} \eta_{n_x} \\ \upsilon_{n_x} \end{bmatrix}\right) = L_f^{n_x} h(\Phi^\dagger(\eta_1, \dots, \eta_{n_x})) + L_g L_f^{n_x-1} h(\Phi^\dagger(\eta_1, \dots, \eta_{n_x}))\upsilon_1. \tag{30}$$

3.6.2 | Case of $n_r < n_x$

Computing $z_1 = y, \dots, z_{n_r} = \frac{d^{n_r-1}}{dt^{n_r-1}} y$ follows as in (27), but

$$z_{n_r+1} = \frac{d^{n_r}}{dt^{n_r}} y = L_f^{n_r} h(x) + L_g L_f^{n_r-1} h(x) u = \Phi_{n_r}(x, u). \tag{31}$$

Continuing the construction of the map gives that

$$z_{n_r+2} = \frac{d}{dt} z_{n_r+1} = L_f^{n_r+1} h(x) + L_g L_f^{n_r} h(x) u + L_f L_g L_f^{n_r-1} h(x) u + L_g^2 L_f^{n_r-1} h(x) u^2 + L_g L_f^{n_r-1} h(x) \frac{d}{dt} u = \Phi_{n_r+1}\left(x, u, \frac{d}{dt} u\right). \tag{32}$$

Repeating this operation recursively results in

$$\frac{d}{dt} z_{n_r+l} = \Phi_{n_r+l-1}\left(x, u, \dots, \frac{d^{l-1}}{dt^{l-1}} u\right), \tag{33}$$

for $1 \leq l \leq n_s + 1$ with $n_s = n_x - n_r - 1$. Compared to the previous case, these maps now depend on $u, \dots \frac{d^{n_s}}{dt^{n_s}} u$. Hence,

$$z = \Phi\left(x, u, \dots, \frac{d^{n_s}}{dt^{n_s}} u\right) = \begin{bmatrix} \Phi_0(x) & \dots & \Phi_{n_r-1}(x) & \Phi_{n_r}(x, u) & \dots & \Phi_{n_x-1}\left(x, u, \dots, \frac{d^{n_s}}{dt^{n_s}} u\right) \end{bmatrix}^\top, \tag{34}$$

and the local inverse of Φ provides Ψ in Definition 3 to construct Γ_{n_x}. We can now state the following lemma presenting the conditions for local invertibility of Φ.

Lemma 2 (Φ inversion under $n_r < n_x$). *Assume full rank of $\nabla\Phi(x_0, u_0, \dots u_0)$, where $\nabla\Phi$ is the Jacobian of Φ w.r.t. x. There exist open sets $x_0 \in \mathbb{X}_0 \subseteq \mathbb{X}_r$, $u_0 \in \mathbb{U}_0 \subseteq \mathbb{R}$, $\mathbb{Y}_0 \subseteq \mathbb{R}$, and an analytic function $\Phi^\dagger : \mathbb{Y}_0^{n_x} \times \mathbb{U}_0^{n_s+1} \to \mathbb{X}_0$, such that for all $\eta \in \mathbb{Y}_0^{n_x}$, $\upsilon \in \mathbb{U}_0^{n_s+1}$ and $x \in \mathbb{X}_0$:*

$$\eta = \Phi(x, \upsilon) \;\Leftrightarrow\; x = \Phi^\dagger(\eta, \upsilon).$$

Lemma 2 follows from the implicit function theorem[44] applied to $\eta - \Phi(x, \upsilon)$. Using $\Phi^{\dagger}$, we can define the function $\Psi : (\mathbb{Y}_0 \times \mathbb{U}_0)^{n_x} \to \mathbb{X}_0$ similarly as in (29) which satisfies Definition 3 by construction. Then, we can proceed with the construction of Γ_{n_x} as in Definition 1, except that Γ_{n_x} will not depend on the last n_r components of $\mathbb{U}^{n_x}$ and hence it can be defined on $\mathbb{Y}_0^{n_x} \times \mathbb{U}_0^{n_s+1}$ instead of $\mathbb{Y}_0^{n_x} \times \mathbb{U}_0^{n_x}$.

The price to pay for a system with relative degree less than its order is that the resulting LPV model through p depends on u and its derivatives up to order n_s. On the other hand, all scheduling signals are potentially directly computable form measured variables without requiring the original states of the system as discussed in Section 3.5.

4 | CONVERSION TO THE FULL OBSERVABILITY FORM

One of the shortcomings of the conversion procedure of Section 3 is that in case of relative degree $n_r < n_x$, the conversion results in an LPV model "depending" on $\left\{\frac{d^l}{dt^l}u\right\}_{l=0}^{n_s}$. This dynamic dependence on u can be undesirable as it increases the complexity of the resulting model. One can say that this is the price to be paid for trying to use only β_0 to express the relations involving u. One way to overcome this is to assume that a part of the state is available for measurement. In that case, parts of x become components of p and they are used to replace the derivatives of u in the dependency structure.

To gain some intuition, consider (9) with a well-defined relative degree $n_r < n_x$ at a point $x = x_0$ and the transformation map Φ from (28). If (u, x, y) is a solution of (9) such that for all $t \in \mathbb{R}$, $x(t) \in \mathbb{X}_r$, then, even in case of $n_r < n_x$, it is possible to use $z(t) = \Phi(x(t))$ as the state of the LPV model, constructed as

$$z = \begin{bmatrix} y & \dots & \frac{d^{n_r-1}}{dt^{n_r-1}}y & L_f^{n_r}h(x) & \dots & L_f^{n_x-1}h(x) \end{bmatrix}. \tag{35}$$

Notice that

$$\frac{d}{dt}z_{n_r+l} = \underbrace{L_f^{n_r+l}h(x)}_{z_{n_r+l-1}} + L_g L_f^{n_r+l-1}h(x)u, \tag{36}$$

for $1 \le l \le n_s + 1$. In terms of Lemma 1, there exist open sets $x_0 \in \mathbb{X}_0 \subseteq \mathbb{X}_r$, $\mathbb{Y}_0 \subseteq \mathbb{R}$, such that $\Phi|_{\mathbb{X}_0} : \mathbb{X}_0 \to \mathbb{Y}_0^{n_x}$ is an analytic diffeomorphism with analytic inverse $\Phi^{\dagger}$. Hence, for $x(t) \in \mathbb{X}_0$, $\forall t \in \mathbb{R}$ and each l, we can write the u-related terms in (36) as

$$L_g L_f^{n_r+l-1}h(x)u = \underbrace{L_g L_f^{n_r+l-1}h(\Phi^{\dagger}(z))}_{\beta_{n_s+1-l}} u. \tag{37}$$

This implies that the last state equation reads as

$$\frac{d}{dt}z_{n_x} = \underbrace{L_f^{n_x}h(\Phi^{\dagger}(z))}_{\Gamma_{n_x}(z)} + \underbrace{L_g L_f^{n_x-1}h(\Phi^{\dagger}(z))}_{\beta_0(z)} u. \tag{38}$$

Intuitively, we want to factorize $\Gamma_{n_x}(z)$, that is, write it as $\Gamma_{n_x}(z) = \sum_{i=0}^{n_x-1}\alpha_i(z)z_{i+1}$. As a result, we obtain Equation (5a–5b) where $\{\alpha_i\}_{i=0}^{n_x-1}$ and $\{\beta_i\}_{i=0}^{n_s}$ are dependent on z. Using that z satisfies (35), which is dependent on $\left\{\frac{d^l}{dt^l}y\right\}_{l=0}^{n_r-1}$ and x, we arrive at an LPV model by taking[#] p as a linear projection of $[y^\top \ x^\top]^\top$.

Now, we present the above procedure more formally. Define the maps $\Gamma_{n_x} : \mathbb{Y}_0^{n_x} \to \mathbb{R}$ and $\{\tilde{\beta}_i : \mathbb{Y}_0^{n_x} \to \mathbb{R}\}_{i=0}^{n_x-1}$ as

$$\Gamma_{n_x}(\zeta) = L_f^{n_x}h(\Phi^{\dagger}(\zeta)), \tag{39a}$$

[#] In theory, it is possible to consider $p = x$. However, this choice results in a scheduling region as large as $\mathbb{X}$ and the resulting LPV model will be overly conservative. Hence, it is a better strategy to include y into p since the derivatives of y and x are closely related. We would then hope that in the final LPV model, most of the state components disappear from p.

$$\tilde{\beta}_i(\zeta) = \begin{cases} L_g L_f^{n_x-i-1} h(\Phi^\dagger(\zeta)) & i \leq n_s; \\ 0 & \text{otherwise}; \end{cases} \tag{39b}$$

for all $\zeta \in \mathbb{Y}_0^{n_x}$. Assume that there exists a set of analytic functions $\{\phi_i\}_{i=1}^{\tau}$ on $\mathbb{Y}_0^{n_x}$ such that the map Γ_{n_x} in (39a) is rational with respect to $\{\phi_i\}_{i=1}^{\tau}$. Let $\left(\{r_i\}_{i=1}^{n_x}, s\right)$ be the factorization of Γ_{n_x} with respect to the first n_x variables. Define the functions $\{\tilde{\alpha}_i : \mathbb{Y}_0^{n_x} \to \mathbb{R}\}_{i=0}^{n_x-1}$ as $\tilde{\alpha}_i = r_{i+1}$. Define $\Psi : \mathbb{Y}_0^{n_r} \times \mathbb{X}_0 \to \mathbb{Y}_0^{n_x}$ as follows

$$\Psi(\eta, x) = \begin{bmatrix} \eta^\top & L_f^{n_r} h(x) & \cdots & L_f^{n_x-1} h(x) \end{bmatrix}^\top, \tag{40}$$

for all $\eta \in \mathbb{Y}_0^{n_r}$, $x \in \mathbb{X}_0$. Notice that $\zeta = \Psi(\eta, x)$, if $(\eta, x) \in \mathbb{V} = \mathbb{Y}_0^{n_r} \times \mathbb{X}_0$. Define now $\hat{\alpha}_i : \mathbb{V} \to \mathbb{R}$ and $\hat{\beta}_i : \mathbb{V} \to \mathbb{R}$ by

$$\hat{\alpha}_i(\eta, x) = \tilde{\alpha}_i(\Psi(\eta, x)), \quad \hat{\beta}_i(\eta, x) = \tilde{\beta}_i(\Psi(\eta, x)),$$

for all $\eta \in \mathbb{Y}_0^{n_r}$, $x \in \mathbb{X}_0$. Let $\mathcal{T}_1 \in \mathbb{R}$ such that for any $x \in \mathbb{X}_0$, $\mathcal{T}_1$ is the selection matrix of the essential support of the functions $\{\eta \mapsto \hat{\alpha}_i(\eta, x), \eta \mapsto \hat{\beta}_i(\eta, x)\}_{i=0}^{n_x-1}$ under $\mathbb{Y}_0$ (essential support w.r.t. the variables $\{\eta_i\}_{i=1}^{n_r}$). Similarly, let $\mathcal{T}_2 \in \mathbb{R}^{m_2 \times n_x}$ with $m_2 \leq n_x$ be the selection matrix of the essential support of the functions $\{\hat{\alpha}_i, \hat{\beta}_i\}_{i=0}^{n_x-1}$ under $\mathbb{X}_0$ w.r.t. x. If $\mathcal{T}_1$ is zero, then let $\mathcal{T} = [0\ \mathcal{T}_2]$ and $\mathcal{P} = [0_{m_2 \times n_r}\ \mathcal{T}_2]$; otherwise, let $\mathcal{T} = \text{diag}(1, \mathcal{T}_2)$ and $\mathcal{P} = \text{diag}(I_{n_r \times n_r}, \mathcal{T}_2)$. Using the notation and assumptions above, we can now state the following theorem.

Theorem 2 (LPV embedding, full observability form). *Under the conditions of Theorem 1, if $s = 0$, that is, factorization of Γ_{n_x} is possible without a remainder, then the LPV-SS representation (8) with coefficient functions $\{\alpha_i := \hat{\alpha}_i \circ \mathcal{P}^{-1},\ \beta_i := \hat{\beta}_i \circ \mathcal{P}^{-1}\}_{i=0}^{n_x-1}$, with*

$$p = \mathcal{T}[y^\top \quad x^\top]^\top, \tag{41}$$

and $\mathbb{P} = \mathcal{T}(\mathbb{Y}_0 \times \mathbb{X}_0)$ satisfies (24c) where

$$\pi_p \mathfrak{B}_{\text{SS}}^{\text{o}} = \left\{(u, x, y) \in C_{n_x}(\mathbb{R}, \mathbb{U}_0 \times \mathbb{X}_0 \times \mathbb{Y}_0) | \exists p \in C_{n_r}(\mathbb{R}, \mathbb{P}), \exists z \in C_{n_x}(\mathbb{R}, \mathbb{Y}_0^{n_x}) \text{ s.t. (5a–b) hold, while } x = \Phi^\dagger(z)\right\}. \tag{42}$$

Proof. The proof of Theorem 2 follows the same line of reasoning as Theorem 1 and hence it is skipped. Note that the construction of Φ implies that $\{\alpha_i \diamond p, \beta_i \diamond p\}_{i=0}^{n_x-1}$ will not depend on the derivative of x. ∎

In contrast with the procedure in Section 3, p here does not include u; however, part of it depends on the availability of the original states of the NL system.

5 | EXAMPLES

First, two academic examples are presented to illustrate the properties of the conversion procedures discussed in the previous sections. In the first one, the relative degree is equal to the order of the system while in the second one, it is less. These examples are followed by the examples of a magnetic levitation system and an unbalanced disc system. In the latter two examples, an NL model derived from first principle laws is converted into an LPV-SS representation. In the last example, we also show empirical validation of the model conversion both in terms of comparing responses of the real system with its LPV model and also how an LPV controller designed based on the converted model performs.

5.1 | Conversion under full relative degree

Consider a SISO NL system model (9) with $n_x = 3$ and

$$f(x) = \begin{bmatrix} 0 \\ x_1 + x_3^2 \\ x_2 + x_2 x_3 \end{bmatrix}, \quad g(x) = \begin{bmatrix} x_2^2 + x_3^2 + 1 \\ 0 \\ 0 \end{bmatrix}, \quad h(x) = x_3.$$

As commonly done in practice, one could pick x_3 and x_2 as scheduling variables for LPV conversion to the form of (5); however, that would require accurate measurements or estimates of these state variables if an LPV controller was to be designed and implemented based on such a converted model. Another problem would be the validity of this LPV conversion in terms of the represented solutions of the original NL model: it would not be clear under which condition the obtained LPV model is a valid representation of the NL model. So, let us see what the proposed method in this article results in. For this system, we have

$$L_g h(x) = 0, \qquad L_g L_f h(x) = 0,$$
$$L_g L_f^2 h(x) = (x_2^2 + x_3^2 + 1)(x_3 + 1),$$

which gives that the relative degree is $n_r = 3 = n_x$ at each x_0 not belonging to the hyperplane $\mathbb{X}_0^\dagger = \{x \in \mathbb{R}^3 | (x_2^2 + x_3^2 + 1)(x_3 + 1) = 0\}$. Select $x_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$ and $\mathbb{X}_r$ to be any open subset of $\mathbb{R}^3 \setminus \mathbb{X}_0^\dagger$. For the sake of simplicity, take $\mathbb{X}_r = (-1, 1)^3$. Computing (28) gives $z = \Phi(x)$ where

$$\Phi(x) = \begin{bmatrix} x_3 & x_2 + x_2 x_3 & (x_3 + 1)(x_2^2 + x_3^2 + x_1) \end{bmatrix}^\top.$$

The Jacobian of Φ is non-singular on $\mathbb{X}_r$, in fact Φ is an analytic diffeomorphism on $\mathbb{X}_r$ and its inverse is given by

$$\Phi^\dagger(\eta) = \begin{bmatrix} \frac{(\eta_3 - (\eta_1+1)^3(\eta_2^2 + \eta_1^2(\eta_1+1)^2))}{(\eta_1+1)} & \frac{\eta_2}{\eta_1+1} & \eta_1 \end{bmatrix}^\top.$$

Let $\mathbb{Y}_0 = (-1, 1)$, which is an open subset of $\mathbb{R}$ and satisfies $\mathbb{Y}_0^3 \subseteq \Phi(\mathbb{X}_r)$ and set $\mathbb{X}_0 = \Phi^\dagger(\mathbb{Y}_0^3)$. Let $\mathbb{U}_0$ be an arbitrary open subset of $\mathbb{R}$ containing 0. The resulting Γ_{n_x} function, see (30), is given by

$$\Gamma_{n_x}(\zeta) = \frac{\eta_2(2\eta_1 + 3\eta_3 + 3\eta_1\eta_3 + 6\eta_1^2 + 6\eta_1^3 - 2\eta_2^2 + 2\eta_1^4) + (\eta_1 + 1)\left(\eta_1(\eta_1+1)^2 + \eta_2^2\right) v_1}{(\eta_1 + 1)^2},$$

where $\zeta = \begin{bmatrix} \eta_1 & v_1 & \dots & \eta_3 & v_3 \end{bmatrix}$. Factorization of this rational function is implemented by applying Algorithm 1 resulting in

$$r_1(\zeta) = 0, \qquad r_2(\zeta) = -\frac{-2\eta_1 + 6\eta_1^2 - 6\eta_1^3 - 2\eta_2^2 + 2\eta_1^4}{(\eta_1+1)^2},$$
$$r_3(\zeta) = \frac{3\eta_2}{\eta_1+1}, \qquad r_4(\zeta) = (\eta_1 + 1)\left(\eta_1 + \frac{\eta_2^2}{(\eta_1+1)^2}\right),$$

with $s = 0$. Hence,

$$\mathcal{R}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathcal{T} = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad \mathcal{T}^{-1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

Then, $\{\alpha_i = r_{i+1} \circ \mathcal{R}^{-1} \circ \mathcal{T}^{-1}\}_{i=0}^2$, and $\beta_0 = r_4 \circ \mathcal{R}^{-1} \circ \mathcal{T}^{-1}$ are defined on $\mathbb{Y}_0^2$ and with the resulting $p = y = \mathcal{T}\begin{bmatrix} y & u \end{bmatrix}^\top$:

$$\alpha_0 \diamond p = 0, \qquad \alpha_1 \diamond p = -\frac{-2p + 6p^2 - 6p^3 + 2p^4 - 2\dot{p}^2}{(p+1)^2},$$
$$\alpha_2 \diamond p = \frac{3\dot{p}}{p+1}, \qquad \beta_0 \diamond p = (p + 1)\left(p + \frac{\dot{p}^2}{(p+1)^2}\right).$$

The scheduling region is $\mathbb{P} = \mathcal{T}(\mathbb{Y}_0 \times \mathbb{U}_0) = \mathbb{Y}_0 = (-1, 1)$. The selection of the scheduling signal $p = y$, leads to the converted LPV model (8) which achieves embedding of the NL behavior into the solution set of the LPV-SS representation according to Theorem 1. It is worth to mention that for this system with $p = y$, the converted matrices have only first order dynamic dependence (dependence on p and $\dot{p}$ only). As a further simplification, in line with Section 3.3, one can introduce $p = \mu \diamond y = \begin{bmatrix} y & \dot{y} \end{bmatrix}^\top$ which results in rational static dependency of $\alpha_0, \alpha_1, \alpha_2, \beta_0$ by increasing the dimension of p, while taking $p = \begin{bmatrix} r_2 \diamond y & r_3 \diamond y & r_4 \diamond y \end{bmatrix}^\top$ results in an affine, but conservative embedding with $\alpha_1 = p_1, \alpha_2 = p_2, \beta_0 = p_3$.

5.2 | Conversion under low relative degree

To demonstrate the properties of the procedures presented in Sections 3 and 4, (9) is considered with $n_x = 3$ and

$$f(x) = \begin{bmatrix} x_2 - 2x_2x_3 + x_3^2 \\ x_3 \\ \sin(x_1) \end{bmatrix}, \quad g(x) = \begin{bmatrix} 4x_2x_3 \\ -2x_3 \\ 0 \end{bmatrix}, \quad h(x) = x_3.$$

The system has a relative degree $n_r = 2 < n_x$ at each x_0 not belonging to the hyper-surface $\mathbb{X}_0^\dagger = \{x \in \mathbb{R}^3 | \cos(x_1)x_2x_3 = 0\}$. Select $x_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$ and let $\mathbb{X}_r = \left(0, \frac{\pi}{2}\right) \times (-1, 1) \times (-0.5, 0.5)$. It is clear that $\mathbb{X}_r$ is an open subset of $\mathbb{R}^3 \setminus \mathbb{X}_0^\dagger$ containing 0. First consider the approach discussed in Section 4 to convert the NL representation to the full observability canonical form (8). According to (28)

$$z = \Phi(x) = \begin{bmatrix} x_3 & \sin(x_1) & \cos(x_1)(x_3^2 - 2x_2x_3 + x_2) \end{bmatrix}^\top.$$

The Jacobian of Φ is non-singular on $\mathbb{X}_r$; in fact, Φ is an analytic diffeomorphism on $\mathbb{X}_r$ and the inverse map is

$$\Phi^\dagger(\zeta) = \begin{bmatrix} \sin^{-1}(\zeta_2) & \frac{-\zeta_3 + \zeta_1^2\sqrt{1-\zeta_2^2}}{(2\zeta_1 - 1)\sqrt{1-\zeta_2^2}} & \zeta_1 \end{bmatrix}^\top.$$

Let $\mathbb{Y}_0^3 \subseteq \Phi(\mathbb{X}_r)$ be an open set and $\mathbb{X}_0 = \Phi^\dagger(\mathbb{Y}_0^3)$. The resulting $\Gamma_{n_x}(\zeta)$ via (39a) is given by

$$\Gamma_{n_x}(\zeta) = \frac{-2\zeta_2\zeta_3 - \zeta_2\zeta_3^2 + 2\zeta_2^3\zeta_3 + 2\zeta_1\zeta_2\zeta_3^2 + \left(4\zeta_1^3 - 4\zeta_1^2 + \zeta_1 + 2\zeta_1\zeta_2 - 2\zeta_1^2\zeta_2\right)\sqrt{(1-\zeta_2^2)^3}}{(2\zeta_1 - 1)(\zeta_2^2 - 1)}, \tag{43}$$

while for all $\zeta \in \mathbb{Y}_0^3$,

$$\tilde{\beta}_0(\zeta) = \frac{-2\zeta_1\left(2\zeta_2\zeta_3^2 - 2\zeta_1^2\zeta_2\zeta_3\sqrt{1-\zeta_2^2}(4\zeta_1^2 - 4\zeta_1 + 1)\sqrt{(1-\zeta_2^2)^3}\right)}{(2\zeta_1 - 1)(\zeta_2^2 - 1)},$$

$$\tilde{\beta}_1(\zeta) = \frac{-4\zeta_1\left(\zeta_3 - \zeta_1^2\sqrt{1-\zeta_2^2}\right)}{2\zeta_1 - 1}, \quad \tilde{\beta}_2(\zeta) = 0.$$

Finally, the factorization step is performed for the function $\Gamma_{n_x}(\zeta)$ via Algorithm 1 as Γ_{n_x} is rational in the considered sense with $\phi_1(\zeta) = \sqrt{1-\zeta_2^2}$, which yields the following functions

$$r_1(\zeta) = \tilde{\alpha}_0(\zeta) = \frac{(4\zeta_1^2 - 4\zeta_1 + 1)\sqrt{1-\zeta_2^2}}{(2\zeta_1 - 1)(\zeta_2^2 - 1)},$$

$$r_2(\zeta) = \tilde{\alpha}_1(\zeta) = \frac{(-2\zeta_1 - 2\zeta_1^2)\sqrt{(1-\zeta_2)^3}}{(2\zeta_1 - 1)(\zeta_2^2 - 1)},$$

$$r_3(\zeta) = \tilde{\alpha}_2(\zeta) = \frac{(-2\zeta_2 - \zeta_2 + 2\zeta_2^3 + 2\zeta_1\zeta_2\zeta_3)}{(2\zeta_1 - 1)(\zeta_2^2 - 1)},$$

with $s=0$. According to (40), computing $\zeta=\Psi(\eta,x)$ gives that $\zeta_1=\eta_1$, $\zeta_2=\eta_2$, $\zeta_3=\sqrt{1-\eta_2^2}(\eta_1^2-2x_2\eta_1+x_2)$ for all $(\eta,x)\in\mathbb{V}=\mathbb{Y}_0^2\times\mathbb{X}_0$. This results in

$$\mathcal{T}_1=1,\quad \mathcal{T}_2=\begin{bmatrix}0 & 1 & 0\end{bmatrix},\quad \mathcal{T}=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix},\quad \mathcal{P}=\begin{bmatrix}1 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 1 & 0\end{bmatrix}.$$

yielding $p=\begin{bmatrix}y & x_2\end{bmatrix}^\top=\mathcal{T}\begin{bmatrix}y & x^\top\end{bmatrix}^\top$ with $\mathbb{P}=\mathcal{T}(\mathbb{Y}_0\times\mathbb{X}_0)$. The resulting coefficients are

$$\alpha_0\diamond p=\frac{2\dot{p}_1\sqrt{1-\dot{p}_1^2}(p_1^2-2p_2p_1+p_2)^2+\left(4p_1^2-4p_1+2\dot{p}_1+1-2p_1\dot{p}_1\right)\sqrt{(1-(\dot{p}_1)^2)^3}}{(2p_1-1)(\dot{p}_1^2-1)}, \tag{44a}$$

$$\alpha_1\diamond p=\frac{(4p_1^2-4p_1+1)\sqrt{1-\dot{p}_1^2}}{(2p_1-1)(\dot{p}_1^2-1)}, \tag{44b}$$

$$\alpha_2\diamond p=\frac{(-2\dot{p}_1-\dot{p}_1+2\dot{p}_1^3+2p_1\dot{p}_1\sqrt{1-\dot{p}_1^2}(p_1^2-2p_2p_1+p_2))}{(2p_1-1)(\dot{p}_1^2-1)}, \tag{44c}$$

$$\beta_0\diamond p=\frac{-4(\dot{p}_1(p_1^3-2p_2p_1+p_2)p_2p_1)-2p_1\sqrt{1-\dot{p}_1^2}(1-2p_2)}{(2p_1-1)(\dot{p}_1^2-1)}, \tag{44d}$$

$$\beta_1\diamond p=\frac{4p_2p_1\sqrt{1-\dot{p}_1^2}}{2p_1-1}, \tag{44e}$$

$$\beta_2\diamond p=0. \tag{44f}$$

where $p\in\mathscr{C}_\infty(\mathbb{R},\mathbb{P})$.

Consider the conversion procedure introduced in Section 3. The map Φ is determined by

$$\Phi(x,u)=\begin{bmatrix}x_3 & \sin(x_1) & \cos(x_1)(x_3^2-2x_2x_3+x_2+4x_2x_3u)\end{bmatrix}^\top.$$

Notice that

$$\nabla\Phi|_{x=0,u=0}=\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix},$$

is full row rank, hence, by Lemma 2, there exist compact open sets $0\in\mathbb{X}_0\subseteq\mathbb{X}_r=(0,\frac{\pi}{2})\times(-1,1)\times(-0.5,0.5)$, $\mathbb{Y}_0\subseteq\mathbb{R}$, $0\in\mathbb{U}_0\subseteq\mathbb{R}$, and an analytic map $\Phi^\dagger:\mathbb{Y}_0^3\times\mathbb{U}_0\to\mathbb{X}_0$, such that $\eta=\Phi(x,\upsilon)\ \Leftrightarrow\ x=\Phi^\dagger(\eta,\upsilon)$ for all $\upsilon\in\mathbb{U}_0,\eta\in\mathbb{Y}_0^3,x\in\mathbb{X}_0$. In this case,

$$\Phi^\dagger(\eta,\upsilon)=\begin{bmatrix}\sin^{-1}(\eta_2) & \frac{-\eta_3+\eta_1^2\sqrt{1-\eta_2^2}}{(2\eta_1-4\eta_1\upsilon-1)\sqrt{1-\eta_2^2}} & \eta_1\end{bmatrix}^\top.$$

According to Corollary 1, Γ_{n_x} with $\zeta=\begin{bmatrix}\eta_1 & \upsilon_1 & \dots & \eta_3 & \upsilon_3\end{bmatrix}$ is given by:

$$\begin{aligned}\Gamma_{n_x}(\zeta)=&\frac{\eta_2\eta_3^2-2\eta_1\eta_2\eta_3^2-2\eta_2\eta_3(\eta_2^2-1)+4\upsilon_1\eta_1\eta_2\eta_3^2+4\upsilon_1\eta_2\eta_3(\eta_2^2-1)+4\upsilon_2\eta_1\eta_3(\eta_2^2-1)}{(\eta_2^2-1)(4\upsilon_1\eta_1-2\eta_1+1)}\\&+\frac{(4\eta_1^2-4\eta_1^3-\eta_1+16\upsilon_1^2\eta_1^2-48\upsilon_1^2\eta_1^3+32\upsilon_1^3\eta_1^3+2\upsilon_1\eta_1-2\eta_1\eta_2-16\upsilon_1\eta_1^2+24\upsilon_1\eta_1^3+4\upsilon_2\eta_1^3+2\eta_1^2\eta_2-4\upsilon_1\eta_1^2\eta_2)\sqrt{(1-\eta_2^2)^3}}{(\eta_2^2-1)(4\upsilon_1\eta_1-2\eta_1+1)}.\end{aligned} \tag{45}$$

Then, the factorization step is performed for Γ_{n_x} with respect to the first 4 variables. Γ_{n_x} is rational in the considered sense with $\phi_1(\zeta) = \sqrt{1-\eta_2^2}$, hence the resulting factorization is $(\{r_i\}_{i=1}^4, s = 0)$, where

$$r_1(\zeta) = \frac{(-4\eta_1 - 4\eta_1^2 + 4\upsilon_2\eta_1^2)\sqrt{(1-\eta_2^2)^3}}{(\eta_2^2 - 1)(4\upsilon_1\eta_1 - 2\eta_1 + 1)}, \tag{46a}$$

$$r_2(\zeta) = \frac{(-2\eta_1 + 2\eta_1^2)\sqrt{(1-\eta_2^2)^3}}{(\eta_2^2 - 1)(4\upsilon_1\eta_1 - 2\eta_1 + 1)}, \tag{46b}$$

$$r_3(\zeta) = \frac{(4\upsilon_2\eta_1 - 2\eta_2)(\eta_2^2 - 1) - 2\eta_2\eta_3 - 2\eta_1\eta_2\eta_3}{(\eta_2^2 - 1)(4\upsilon_1\eta_1 - 2\eta_1 + 1)}, \tag{46c}$$

$$r_4(\zeta) = \frac{4\eta_1\eta_2\eta_3^2 + 4\eta_1\eta_3(\eta_2^2 - 1) + \sqrt{(1-\eta_2^2)^3}(-4\eta_1^2\eta_2 + 24\eta_1^3 - 16\eta_1^2 + 2\eta_1 + 32\upsilon_1^2\eta_1^3 - 48\upsilon_1\eta_1^2 - 16\upsilon_1\eta_1^2)}{(\eta_2^2 - 1)(4\upsilon_1\eta_1 - 2\eta_1 + 1)}, \tag{46d}$$

which holds for all $\zeta \in (\mathbb{Y}_0 \times \mathbb{U}_0)^3$ and

$$\mathcal{T} = \mathcal{T}^{-1} = I, \quad \mathcal{R}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

due to the full joint essential support of $\{r_i \circ \mathcal{R}^{-1}\}_{i=1}^{n_x}$. The system is embedded into the LPV-SS form (8), as described in Theorem 1, with $\mathbb{P} = \mathbb{Y}_0 \times \mathbb{U}_0$ and $\{\alpha_i\}_{i=0}^3$, β_0 satisfying

$$\alpha_0 \diamond p = \frac{(-4p_1 - 4p_1^2 + 4\dot{p}_2p_1^2)\sqrt{(1-\dot{p}_1^2)^3}}{(\dot{p}_1^2 - 1)(4p_2p_1 - 2p_1 + 1)}, \tag{47a}$$

$$\alpha_1 \diamond p = \frac{(-2p_1 + 2p_1^2)\sqrt{(1-\dot{p}_1^2)^3}}{(\dot{p}_1^2 - 1)(4p_2p_1 - 2p_1 + 1)}, \tag{47b}$$

$$\alpha_2 \diamond p = \frac{(4\dot{p}_2p_1 - 2\dot{p}_1)(\dot{p}_1^2 - 1) - 2\dot{p}_1\ddot{p}_1 - 2p_1\dot{p}_1\ddot{p}_1}{(\dot{p}_1^2 - 1)(4p_2p_1 - 2p_1 + 1)}, \tag{47c}$$

$$\beta_0 \diamond p = \frac{4p_1\dot{p}_1\ddot{p}_1^2 + 4p_1\ddot{p}_1(\dot{p}_1^2 - 1) + \sqrt{(1-\dot{p}_1^2)^3}(-4p_1^2\dot{p}_1 + 24p_1^3 - 16p_1^2 + 2p_1 + 32p_2^2p_1^3 - 48p_2p_1^2 - 16p_2p_1^2)}{(\dot{p}_1^2 - 1)(4p_2p_1 - 2p_1 + 1)}, \tag{47d}$$

for all $p \in \mathscr{C}_\infty(\mathbb{R}, \mathbb{P})$. Note that the resulting LPV-SS model has 2nd-order dynamic dependency on $p_1 = y$ and only static dependency on $p_2 = u$. Furthermore, $\mathbb{P}$ can be chosen to be any open subset of $\{(y, u) \in \mathbb{R} \times \mathbb{R} \mid (y^2 - 1)(4uy - 2y + 1) \neq 0\}$.

5.3 | Magnetic levitation system

To show how the proposed methodology performs in practical applications, consider a magnetic levitation system, discussed in Reference 45, which consists of an iron ball, an electromagnet and a photo diode based position sensor. The iron ball is levitated by the attractive force of the electromagnet, which is controlled by an applied voltage (input signal).

TABLE 1 Physical parameters of the magnetic levitation system

L [H]	*R* [Ohm]	*M* [kg]	*G* [m/s²]	*δ* [m]	*Q* [Hm]
2.05	27.03	0.357	9.807	0.0078	0.0044

The model of the system can be represented in the form of (9) with

$$f(x) = \begin{bmatrix} x_2 \\ G - \frac{Qx_3^2}{2M(\delta+x_1)^2} \\ \frac{x_3(2\delta+x_1)(Qx_2-R(\delta+x_1)^2)}{(\delta+x_1)\left((L+Q)(2\delta+x_1)+Q\right)} \end{bmatrix}, \; g(x) = \begin{bmatrix} 0 \\ 0 \\ \frac{(\delta+x_1)(2\delta+x_1)}{(L+Q)(2\delta+x_1)+Q} \end{bmatrix},$$

and $h(x) = x_1$ corresponding to $n_x = 3$ together with the parameter values given in Table 1. The control objective for this system is to keep the distance x_1 (the output signal) of the ball from the magnet close to some level $\delta_{\min} \le \delta \le \delta_{\max}$, where $\delta_{\min} > 0$ corresponds to the minimal distance of the ball from the magnet, while $\delta_{\max}$ corresponds to the maximum allowed height of levitation. The system has a relative degree $n_r = 3 = n_x$ at each x_0 not belonging to the hyperplane $\mathbb{X}_0^\dagger = \{x \in \mathbb{R} | \delta + x_1 = 0\}$. Note that this is physically always satisfied as x_1 must be positive otherwise the ball reaches the magnet plate. Take $\mathbb{X}_r = (\delta_{\min}, \delta_{\max})^3$ and select $x_0 = \begin{bmatrix} \delta & 0 & 2\delta\sqrt{\frac{2GM}{Q}} \end{bmatrix}^\top$. Then, (28) is of the form

$$\Phi(x) = \begin{bmatrix} x_1 & x_2 & G - \frac{Qx_3^2}{2M(\delta+x_1)^2} \end{bmatrix}^\top.$$

The Jacobian of Φ is non-singular on $\mathbb{X}_r$:

$$\nabla\Phi|_{x=x_0} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{G}{\delta} & 0 & -\sqrt{\frac{GQ}{2M\delta^2}} \end{bmatrix}.$$

Hence, there exist open sets $x_0 \in \mathbb{X}_0 \subseteq \mathbb{X}_r$, $\mathbb{Y}_0 \subseteq \mathbb{R}$, such that $\Phi(\mathbb{X}_0) = \mathbb{Y}_0^3$ and the restriction of Φ to $\mathbb{X}_0$ is an analytic diffeomorphism. The inverse map $\Phi^\dagger : \mathbb{Y}_0^3 \to \mathbb{X}_0$ is

$$\Phi^\dagger(\eta) = \begin{bmatrix} \eta_1 & \eta_2 & (\delta+\eta_1)\sqrt{\frac{2M(G-\eta_3)}{Q}} \end{bmatrix}^\top,$$

for all $\eta \in \mathbb{Y}_0^3$. The resulting function Γ_{n_x} with $\zeta = \begin{bmatrix} \eta_1 & \upsilon_1 & \dots & \eta_3 & \upsilon_3 \end{bmatrix}$, see (30), is

$$\Gamma_{n_x}(\zeta) = \frac{2(G-\eta_3)\left(R(\delta+\eta_1)^2(2\delta+\eta_1) + \eta_2\left(Q + L(2\delta+\eta_1)\right)\right)}{(\delta+\eta_1)(L(2\delta+\eta_1) + Q(1+2\delta+\eta_1))} - \frac{(2\delta+\eta_1)\sqrt{2Q(G-\eta_3)}}{\sqrt{M}(Q+(2\delta+\eta_1)(L+Q))}\upsilon_1. \tag{48}$$

Then Γ_{n_x} is rational in the considered sense with $\phi_1(\zeta) = \sqrt{G-\eta_3}$ and Algorithm 1 yields the factorization $(\{r_i\}_{i=1}^4, s)$:

$$r_1(\zeta) = \frac{2GR\eta_1^2 + 8GR\delta\eta_1 + 10GR\delta^2}{(\delta+\eta_1)(L(2\delta+\eta_1) + Q(1+2\delta+\eta_1))},$$

$$r_2(\zeta) = -\frac{2GQ + 4GL\delta + 2GL\eta_1}{(\delta+\eta_1)(L(2\delta+\eta_1) + Q(1+2\delta+\eta_1))}$$

$$r_3(\zeta) = \frac{-2Q\eta_2 - 2R\eta_1^3 - 4R\delta^3 - 2L\eta_1\eta_2 - 8R\delta\eta_1^2 - 10R\delta^2\eta_1 - 4L\delta\eta_2}{(\delta+\eta_1)(L(2\delta+\eta_1) + Q(1+2\delta+\eta_1))},$$

$$r_4(\zeta) = -\frac{(2\delta+z_1)\sqrt{2Q(G-\eta_3)}}{\sqrt{M}(Q+(2\delta+\eta_1)(L+Q))},$$

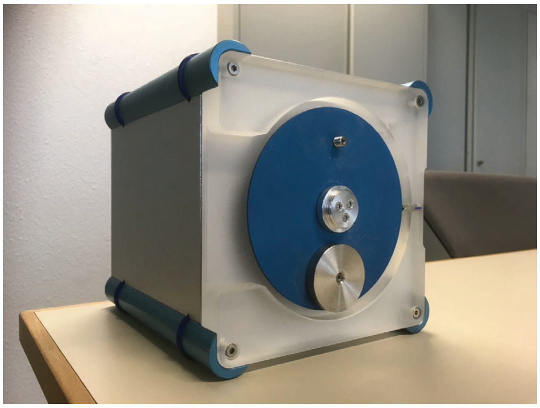


FIGURE 3 Unbalanced disc system: DC motor connected to a disc with added weight. The overall system functions as a rotational pendulum

TABLE 2 Identified parameters of the unbalanced disc system

g [m/s^2]	J [kg·m^2]	K_m [rad/Vs^2]	l [m]	M [kg]	τ [1/s]
9.8	$2.4 \cdot 10^{-4}$	11	0.041	0.076	0.40

with a non-factorizable term given by

$$s(\zeta) = \frac{4GR\delta^3}{(\delta+\eta_1)(L(2\delta+\eta_1)+Q(1+2\delta+\eta_1))}.$$

Therefore, the LPV representation (8) for the system can be obtained, where $p = y$ with 2nd order dynamic dependence $\left(\text{dependence on } \left\{\frac{d^i}{dt^i}y\right\}_{i=0}^{2}\right)$ and the non-factorizable term can be handled by seeing it as a *virtual input*, see Section 3.4.

5.4 | Unbalanced disc system

As an additional example, we demonstrate empirically the applicability of the proposed method. Consider the unbalanced disc system depicted in Figure 3. The dynamic behavior of this system can be well described using the following motion equations where the fast electrical subsystem is neglected

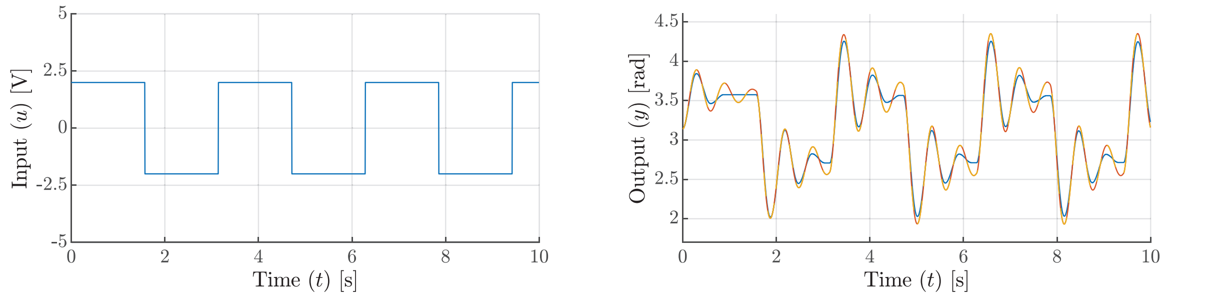
$$\dot{\theta}(t) = \omega(t), \tag{49a}$$

$$\dot{\omega}(t) = \frac{Mgl}{J}\sin(\theta(t)) - \frac{1}{\tau}\omega(t) + \frac{K_m}{\tau}u(t), \tag{49b}$$

where θ is the angular position of the mass, ω is the angular velocity of the mass, and u is the applied voltage on the motor. Note that θ is measurable via an encoder and it corresponds to the output of the plant. The physical parameters of (49) have been estimated based on measurement data collected with a sampling time of $t_\mathrm{s} = 0.01$ s and are given in Table 2. By comparing the simulated response of the nonlinear model (using `ode8` in MATLAB with fixed step-size t_s) and the real system for a voltage signal profile that was not used in the estimation data set, we can observe from Figure 4 that (49) with the estimated parameters successfully captures the physical dynamics with a *best fit rate* (BFR)[‖] of 98.0%. Further details of the parameter estimation and the involved measurement signals can be found in Reference 46.

[‖]BFR is an error measure used to compare data samples $y(k)$ (N data points) w.r.t. an approximation $\hat{y}(k)$, for example, y is the measurement data and $\hat{y}$ is the response of the NL/LPV model. The BFR is computed as

$$\mathrm{BFR}(y,\hat{y}) := \max\left(1 - \frac{\sum_{k=1}^{N}(y(k)-\hat{y}(k))^2}{\sum_{k=1}^{N}(y(k)-\mathrm{mean}(y))^2}, 0\right). \tag{50}$$



(A) Input signal u (B) Output responses — measurement, — nonlinear model, - - LPV model

FIGURE 4 Empirical validation of the identified nonlinear model and the converted LPV model

By reformulating (49) in terms of a SISO NL state-space model (9) with $x = \begin{bmatrix}\theta & \omega\end{bmatrix}^\top$ and

$$f(x) = \begin{bmatrix} x_2 \\ \frac{Mgl}{J}\sin(x_1) - \frac{1}{\tau}x_2 \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ \frac{K_m}{\tau} \end{bmatrix}, \quad h(x) = x_1,$$

we can apply the procedures presented in Section 3 to obtain an LPV model of the system.

In this case, $L_g h(x) = 0$ and $L_g L_f h(x) = \frac{K_m}{\tau}$ which gives that the relative degree is $n_\mathrm{r} = 2 = n_\mathrm{x}$ on $\mathbb{R}^2$. Select $x_0 = \begin{bmatrix}0 & 0\end{bmatrix}^\top$ and, for the sake of simplicity, $\mathbb{X}_\mathrm{r} = (-\pi, \pi)^2$. Computing (28) gives $z = \Phi(x) = \begin{bmatrix}x_1 & x_2\end{bmatrix}^\top$, which is an analytic diffeomorphism with $\Phi^\dagger(\eta) = \begin{bmatrix}\eta_1 & \eta_2\end{bmatrix}^\top$. Let $\mathbb{Y}_0 = (-\pi, \pi)$, which satisfies $\mathbb{Y}_0^2 = \Phi(\mathbb{X}_\mathrm{r})$ and set $\mathbb{X}_0 = \Phi^\dagger(\mathbb{Y}_0^2) = \mathbb{X}_\mathrm{r}$. Let $\mathbb{U}_0$ be an arbitrary open subset of $\mathbb{R}$ containing 0. The resulting Γ_{n_x} function, see (30), is given by

$$\Gamma_{n_\mathrm{x}}(\zeta) = \frac{Mgl}{J}\sin(\eta_1) - \frac{1}{\tau}\eta_2 + \frac{K_m}{\tau}\upsilon_1,$$

where $\zeta = \begin{bmatrix}\eta_1 & \upsilon_1 & \eta_2 & \upsilon_2\end{bmatrix}$. This function is polynomial with $\phi_1(\zeta) = \frac{\sin\eta_1}{\eta_1} = \mathrm{sinc}(\eta_1)$, and applying Algorithm 1 results in

$$r_1(\zeta) = \frac{Mgl}{J}\mathrm{sinc}(\eta_1), \qquad r_2(\zeta) = -\frac{1}{\tau}, \qquad r_3(\zeta) = \frac{K_m}{\tau},$$

with $s = 0$. Hence, choosing $p = \mathrm{sinc}(y)$:

$$\alpha_0 \diamond p = \frac{Mgl}{J}p, \qquad \alpha_1 \diamond p = -\frac{1}{\tau}, \qquad \beta_0 \diamond p = \frac{K_m}{\tau}.$$

The scheduling region is $\mathbb{P} = \mu(\mathbb{Y}_0) = \mathbb{Y}_0 = (-0.22,\ 1)$. The selection of the scheduling signal $p = \mathrm{sinc}(y)$, leads to the converted LPV model (8) with affine static dependency that achieves embedding of the NL behavior into the solution set of the LPV-SS representation according to Theorem 1. To summarize, the NL system (49) is embedded in the LPV representation

$$\underbrace{\begin{bmatrix}\dot{\theta}(t) \\ \dot{\omega}(t)\end{bmatrix}}_{\dot{x}(t)} = \begin{bmatrix} 0 & 1 \\ \frac{Mgl}{J}p(t) & -\frac{1}{\tau}\end{bmatrix} \underbrace{\begin{bmatrix}\theta(t) \\ \omega(t)\end{bmatrix}}_{x(t)} + \begin{bmatrix}0 \\ \frac{K_m}{\tau}\end{bmatrix} u(t), \tag{51a}$$

$$y(t) = \begin{bmatrix}1 & 0\end{bmatrix} x(t), \tag{51b}$$

where $p(t) = \mathrm{sinc}(\theta(t))$ with $\mathbb{P} = (-0.22,\ 1)$. By comparing the response[**] of (51), displayed in Figure 4, with the measurements and the simulated response of the NL model, it is apparent that the LPV model response is identical to the NL model simulation.

[**] As the NL model is unstable, the simulated response of (51) is based on p computed from the output of the NL simulation model.

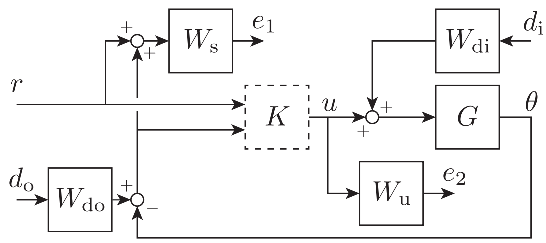


FIGURE 5 Two degree of freedom control structure with mixed-sensitivity shaping for controller synthesis with the LPV model of the unbalanced disc system

A remaining question to be answered is that the resulting LPV model can be used to obtain a high-performance controller of the unbalanced disc system. For this purpose, a two degree of freedom control structure with mixed-sensitivity shaping is considered, depicted in Figure 5, where d_i is an input disturbance, d_o an output disturbance, and r is the reference trajectory which act as disturbances to the resulting generalized plant. Furthermore, $\{e_i\}_{i=1}^2$ in terms of tracking error and control input are the performance channels. The weighting filters are chosen as

$$\begin{aligned} W_s(s) &= \tfrac{0.5012s+2.005}{s+0.02005}, \qquad W_u(s) = \tfrac{s+40}{s+4000}, \\ W_{di} &= 0.5, \qquad W_{do} = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}. \end{aligned} \tag{52}$$

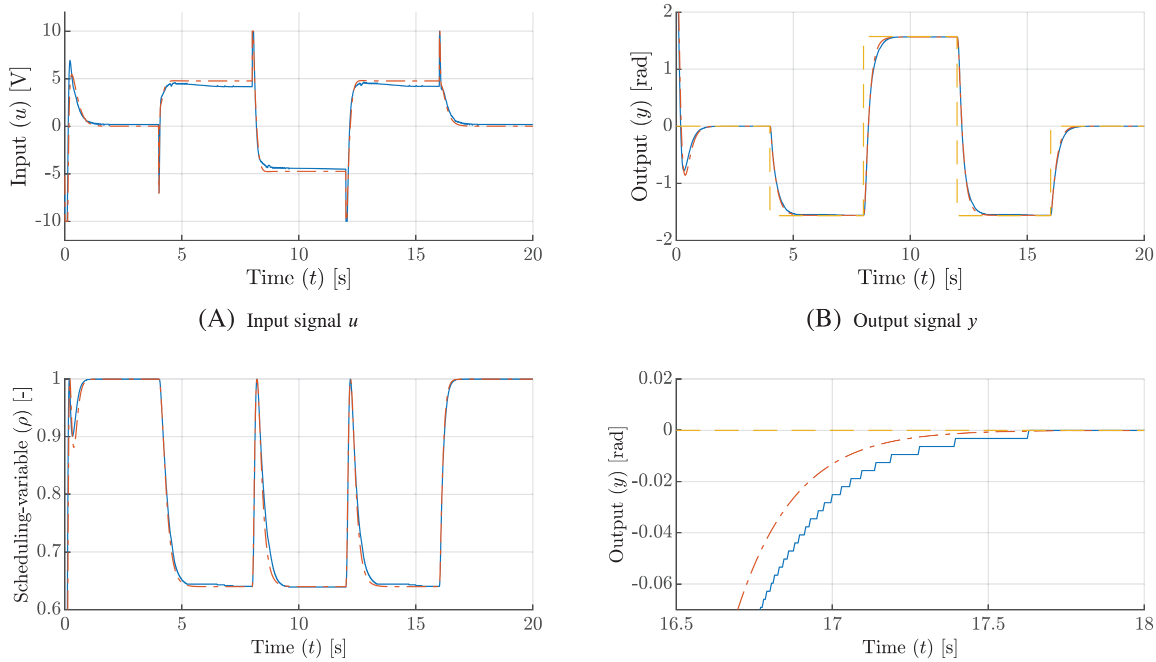
Synthesis of an LPV controller by minimizing the $\mathcal{L}_2$ gain of the disturbance to performance transfer in the shaped generalized plant has been solved using polytopic synthesis based on Reference 47. The resulting controller achieves an $\mathcal{L}_2$ bound of 0.56, that is, it successfully realizes the weighting filters encoded performance objectives. Testing the tracking capabilities of the LPV controller with the NL model (49) in simulation using a reference signal is displayed in Figure 6. The controller provides a smooth reference tracking of the NL closed-loop system with a BFR of 81.7%. The controller was also implemented on the real system and the measured closed-loop response is displayed in Figure 6. The achieved tracking performance[††] in terms of BRF is 82.0%. This proves that the proposed LPV modeling method can be successfully applied to design an LPV controller for a nonlinear system with desired stability and performance guarantees.

5.5 | Distillation column system

As a final example, we show how higher order derivatives of measured output signals involved in the scheduling map can be handled in the implementation of LPV controllers designed based on our LPV model conversion method. Consider the NL first principles-based model of a 4-stage binary distillation column as described in details in Reference 48. Distillation columns are commonly used in the chemical industry for component separation of liquid mixtures based on the differences in the volatility (i.e., boiling point) of the components. The output of the system considered here is the mole fraction of the most volatile component of the distillate product and the input is the inflow rate of the liquid to be separated. The model is represented by (9) with

$$f(x) = \frac{1}{M}\begin{bmatrix} q_F F(x_2 - x_1) - V(\delta(x_1) - x_1) \\ -q_F F(x_2 - x_3) + V(\delta(x_1) - \delta(x_2)) \\ z_F F - q_F F x_3 - (1 - q_F) F\delta(x_3) + V(\delta(x_2) - \delta(x_3)) \\ (1 - q_F) F(\delta(x_3) - x_4) + V(\delta(x_4) - x_4) \end{bmatrix}, \quad g(x) = \frac{1}{M}\begin{bmatrix} x_2 - x_1 \\ x_3 - x_2 \\ x_4 - x_3 \\ 0 \end{bmatrix}, \quad h(x) = x_4. \tag{53}$$

[††]The performance increase w.r.t. to the simulation is due to the inaccuracy of the identified NL model and in other applications such inaccuracies can result in performance changes as with any other model based approach.



(A) Input signal u

(B) Output signal y

(C) Scheduling varaible p

(D) Zoomed-in response of y

FIGURE 6 Closed-loop response with the LPV controller: — experiment, -·- simulation, - - reference

TABLE 3 Physical parameters of the distillation column system

M [kmol]	z_F [mole frac.]	F [kmol/min]	q_F	V [kmol/min]
30	0.65	215	1.0	1800

corresponding to $n_x = 4$ and $\delta(x_i)$ defined as

$$\delta(x_i) = \frac{\tau x_i}{(\tau - 1)x_i + 1}, \quad i = 1, 2, 3, 4,$$

where each x_i stands for the mole fraction of the most volatile component (light component) in the liquid phase on tray i. The values of the physical/chemical parameters in (53) are given in Table 3 with $\tau = 1.2$. The system has a relative degree $n_r = 2 < n_x$ for all $x \in \mathbb{X}^4$. Therefore, the LPV conversion can be performed by the method introduced in Section 3. Note that the method of Section 4 is infeasible in a realistic application of a distillation column, as the states represent concentration levels of the liquid phase on each tray which are impossible to be accurately measured online. Hence, the procedure of Section 3 is applied. The map Φ of the form (34), its inverse $\Phi^\dagger$ and the sets $\mathbb{X}_0, \mathbb{U}_0, \mathbb{Y}_0$ have been computed according to Lemma 2, and used to compute Γ_{n_x}. The latter is used to transform the original NL model to the LPV model in the form (8) by factorizing the term Γ_{n_x} using Algorithm 1. The resulting scheduling dependence is a 3rd-order dynamic dependence on $p = \begin{bmatrix} y & u \end{bmatrix}^\top$. The exact forms of the resulting Γ_{n_x} and the factorized coefficients are not given here due to the lack of space.

Next, we validate the applicability of LPV control based on the obtained equivalent LPV representation when noisy output measurements are considered. To provide a realistic control scenario that respects the involved constraints of the system, we apply an LPV *model predictive control* (MPC)[49] method. MPC algorithms compute an optimal control input at each discrete time instant k by solving an optimization problem based on a prediction model of the process and a cost function characterizing the performance goal (e.g., reference tracking). For this purpose, an accurate model of the process is crucial for the success of such a control methodology. The main advantage of the LPV formulation of the MPC problem is that in general it offers convex optimization based solution by trading off performance due to conservatism of the prediction model.

Based on the derived LPV representation of (53), we can use directly the converted state in the MPC problem, which is composed of the output of the system and its derivatives up to 3rd-order. However, the challenge here is that we need the derivatives of the output (up to order 3), which can be obtained by numerical differentiation and hence the measurement noise can be significantly amplified, affecting the overall performance of the closed-loop system. We also use this converted state and the input together with its derivatives up to order 2 to compute the scheduling variable p, which is used to update the parameter-dependent system matrices of the prediction model at every sampling time. The exact implementation is explained later.

The optimization problem of the MPC considered here is formulated as follows

$$\min_{\Delta u(0|k),\Delta u(1|k),\ldots,\Delta u(N|k)} \sum_{i=0}^{N} (r(i|k) - y(i|k))^{\top} Q (r(i|k) - y(i|k)) + \Delta u^{\top}(i|k) R \Delta u(i|k), \tag{54a}$$

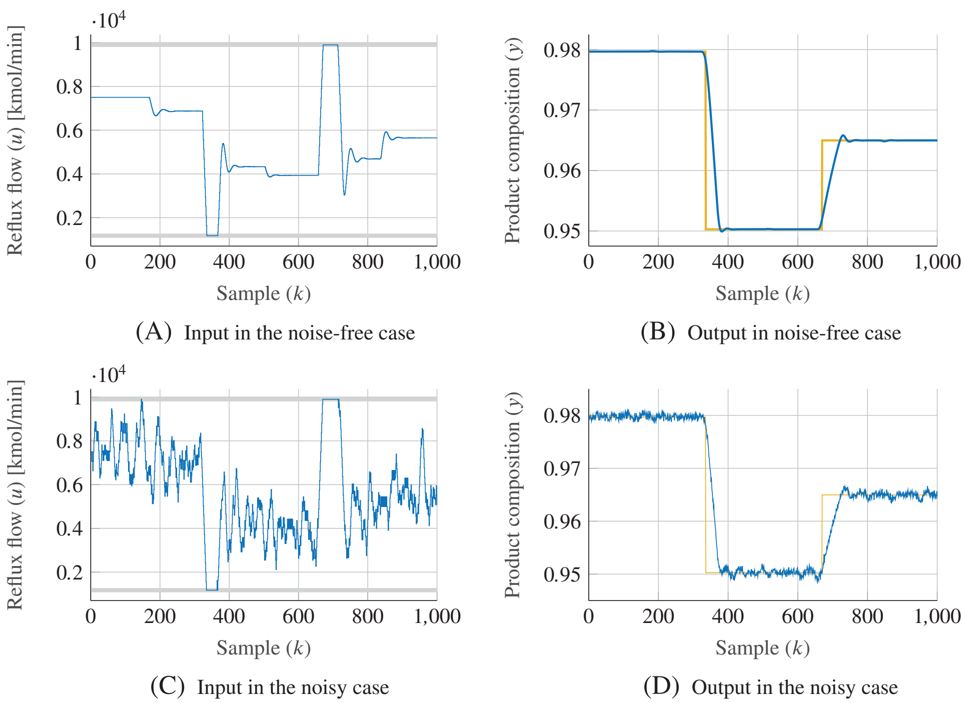
$$\text{s.t.} \quad \Delta u_{\min} \leq \Delta u(i|k) \leq \Delta u_{\max}, \tag{54b}$$

$$u_{\min} \leq u(i|k) \leq u_{\max}, \tag{54c}$$

$$y_{\min} \leq y(i|k) \leq y_{\max}, \tag{54d}$$

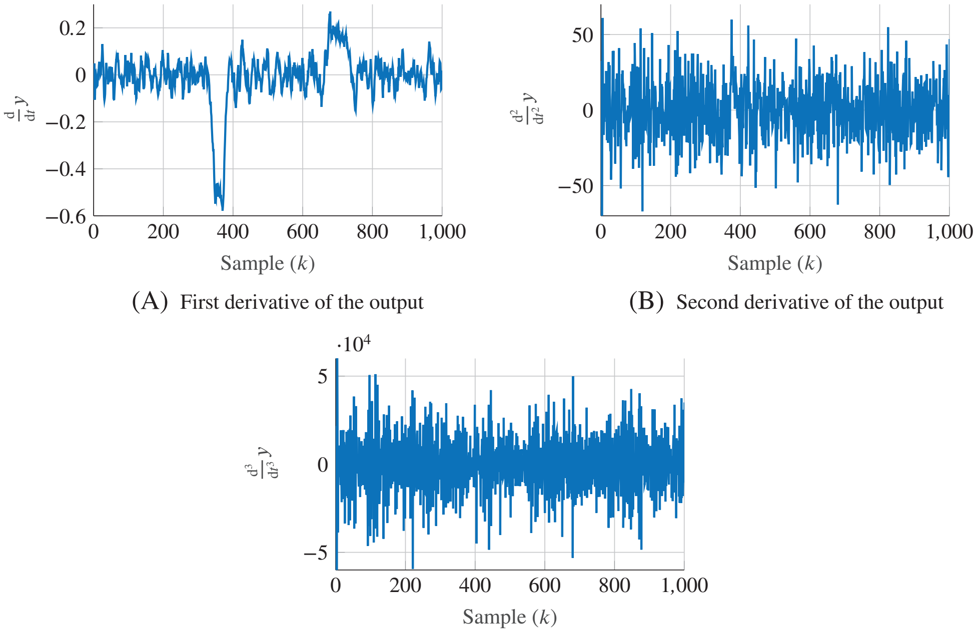
$i = 0, 1, \ldots, N$, where the argument $i|k$ indicates prediction step i at instant k, r is the reference trajectory, Δu represents the rate of change of u, N is the prediction horizon and $Q \geq 0$, $R > 0$ are tuning matrices. The decision variable of the optimization problem (54) is Δu, and hence, we can achieve offset-free control. In order to realize such an MPC scheme, we discretized the obtained continuous-time LPV model using the Euler's forward method, considered Δu as the rate of change of the reflux, and as an output y the purity of the top product was used. For constraints, we considered $[\Delta u_{\min},\ \Delta u_{\max}] = [-436.25, +436.25]$, $[u_{\min},\ u_{\max}] = [1175,\ 9900]$ kmol/min, and $[y_{\min},\ y_{\max}] = [0.85,\ 0.99]$ for $\Delta u, u, y$, respectively. The prediction horizon of the MPC has been taken as $N = 15$, and we consider the weights of the output and the input in the MPC cost function, which is quadratic, as $Q = 10^7$ and $R = 10^{-5}$, respectively. The MPC online optimization problem (54) is cast as a quadratic programming problem.

The performance of the closed-loop system with the LPV MPC has been evaluated with −3% change in the set point of y, at the sampling instant $k = 334$ followed by +1.5% change in the set point at $k = 668$ as shown in Figure 7. At the



(A) Input in the noise-free case
(B) Output in noise-free case
(C) Input in the noisy case
(D) Output in the noisy case

FIGURE 7 Closed-loop response of the distillation column system with the LPV MPC controller: — limits, — simulation, — reference



(A) First derivative of the output

(B) Second derivative of the output

(C) Third derivative of the output

FIGURE 8 The filtered derivative of scheduling signals used to update the distillation column prediction model for the LPV MPC implementation

same time, we have applied three changes of the feed flow rate F as input disturbances: a −20% decrease at $k = 167$, again a −20% decrease at $k = 501$ and a +40% increase at $k = 835$. Such scenario of operation is similar to what was discussed in Reference 50. For comparison, we carried out the simulation for two cases, with noisy and noise-free output. In case of the noisy output, a signal-to-noise ratio of 29.5 dB has been considered with additive white Gaussian measurement noise. To reduce the noise effects in the numerically differentiated signals, which include $\frac{\mathrm{d}}{\mathrm{d}t}y$, $\frac{\mathrm{d}^2}{\mathrm{d}t^2}y$, and $\frac{\mathrm{d}^3}{\mathrm{d}t^3}y$, we used moving average filters of order 10, 2, and 2, respectively. The orders were chosen to find a suitable trade-off between noise filtering, truncation of the frequency content and introduced phase lag. The output derivatives are recursively filtered and used to construct the model represented state variables at every sample. They are used also together with the input and its derivatives $\frac{\mathrm{d}}{\mathrm{d}t}u$ and $\frac{\mathrm{d}^2}{\mathrm{d}t^2}u$ to compute p and hence to update the LPV model matrices during the MPC implementation.

Based on the above discussed discrete-time implementation[‡‡] of the MPC controller, the closed-loop system has been simulated with the plant dynamics taken as the continuous-time NL model in (53) with synchronized ZOH actuation and sampling. Figure 7A–D show the closed-loop performance with and without output measurement noise. Generally, the effect of the noise increases the fluctuation of the applied u and slightly y; however, the tracking capability is still comparable to the case of noise-free y. In both cases, the desired set points of the output are reached within less than 50 samples with almost no overshoot and no steady-state error. The disturbance effects are successfully rejected in both cases by the MPC design. The filtered derivatives of the output y, which are used as scheduling signals for updating the distillation column prediction model during the MPC implementation, are shown in Figure 8A–C.

Finally, to measure numerically the effects of the noise on the control performance, the mean square tracking errors with and without measurement noise were calculated to be 1.47×10^{-5} and 1.40×10^{-5}, respectively. The quadratic cost of the MPC optimization can be seen as a performance measure, for which the average cost with and without measurement noise was 2.37×10^3 and 1.934×10^3, respectively. The cost is larger for the noisy case by a factor of 1.22,

[‡‡]Note that objective of this example is to demonstrate applicability of the conversion scheme in case of process systems where construction of p leads to high order derivatives of y, while significant noise also affects the output measurements. For this purpose, the applied MPC scheme required discretization of the converted continuous-time LPV model. Hence, one can argue if discretization of the NL model first and conversion of it to an LPV form could lead to a more simple computation of p. As the applied discretization scheme affects the overall scheduling construction and the performance of the resulting closed-loop and due to the lack of extension of the proposed methodology for the discrete-time case, answering such a question is beyond the scope of the current article.

which indicates that performance degradation is not significant due to the measurement noise. Finally, we repeated the simulation with lower values of signal-to-noise ratio (SNR) but with the same tuning parameters Q, R, N and filters as above and with the same seed settings for the noise generator. For an SNR of 23.5 dB, the mean square tracking error and the average cost were 1.63×10^{-5} and 3.42×10^{3}, respectively, which still indicate reasonable performance; however, below that value of SNR, it was necessary to tune Q, R, N, to avoid infeasibility of the MPC optimization problem.

In summary, this example demonstrates that reasonable closed-loop performance can be achieved with the proposed method using high-order output derivatives with noisy measurements in the scheduling map without the need of direct state measurements or nonlinear observers designed for the process.

6 | CONCLUSIONS AND FUTURE WORKS

In this article, a systematic and automated approach has been introduced to synthesize LPV state-space representations of nonlinear systems via the idea of multi-path feedback linearization. The main advantage of the proposed approach is its ability to synthesize the model with minimal scheduling dependency where the scheduling map is based on only measurable input-output signals of the original system. This ensures implementability and minimized conservativeness of the LPV embedding. However, as demonstrated by the procedure, this often results in dynamic dependency over these signals. To avoid dynamic dependency especially over input variables, a modified version of the approach is presented that substitutes those dependencies with dependency relation on only part of the state variables of the original nonlinear representation.

CONFLICT OF INTEREST

The authors declared that they have no conflict of interest to this work.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Hossam S. Abbas https://orcid.org/0000-0002-5264-5906
Roland Tóth https://orcid.org/0000-0001-7570-6129
Nader Meskin https://orcid.org/0000-0003-3098-9369
Patrick J.W. Koelewijn https://orcid.org/0000-0003-2378-6264

REFERENCES

1. Scherer CW. Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control for time-varying and linear parametrically-varying systems. *Int J Robust Nonlinear Control*. 1996;6(9-10):929-952.
2. Mohammadpour J, Scherer CW. *Control of Linear Parameter Varying Systems with Applications*. Springer-Verlag; 2012.
3. Besselmann T, Löfberg J, Morari M. Explicit MPC for LPV systems: stability and optimality. *IEEE Trans Automat Contr*. 2012;57(9):2322-2332.
4. Wollnack S, Abbas HS, Werner H, Tóth R. Fixed-structure LPV controller synthesis based on implicit input-output representations. *Automatica*. 2017;83:282-289.
5. Abbas HS, Hanema J, Tóth R, Meskin N, Mohammadpour J. An improved robust model predictive control for linear parameter-varying input-output models. *Int J Robust Nonlinear Control*. 2018;28:859-880.
6. Tóth R. *Modeling and Identification of Linear Parameter-Varying Systems*. Lecture Notes in Control and Information Sciences. Vol 403. Springer; 2010.
7. Rugh W, Shamma JS. Research on gain scheduling. *Automatica*. 2000;36(10):1401-1425.
8. Bachnas AA, Tóth R, Mesbah A, Ludlage J. A review on data-driven linear parameter-varying modeling approaches: a high-purity distillation column case study. *J Process Control*. 2013;24:272-285.
9. Petersson D, Löfberg J. Identification of LPV state-space models using $\mathcal{H}_2$ minimisation. *Optimization Based Clearance of Flight Control Laws*. Springer; 2012:111-128.
10. Shamma JS, Athans M. Analysis of gain scheduled control for nonlinear plants. *IEEE Trans Automat Contr*. 1990;35(8):898-907.
11. Bruzelius F, Pettersson S, Breitholtz C. Linear parameter-varying descriptions of nonlinear systems. Proceedings of the American Control Conference; 2004:1374-1379; Boston, MA.

12. Isidori A. *Nonlinear Control Systems: An Introduction*. Lecture Notes in Control and Information Sciences. Springer; 1995.
13. Bianchi FD, De Battista H, Mantz RJ. *Wind Turbine Control Systems; Principles, Modeling and Gain Scheduling Design*. Springer-Verlag.; 2007.
14. Rugh WJ. Analytical framework for gain scheduling. *IEEE Control Syst Mag*. 1991;11(1):79-84.
15. Shamma JS, Cloutier JR. Gain-scheduled missile autopilot design using linear-parameter varying transformations. *AIAA J Guid Control Dyn*. 1993;16(2):256-263.
16. Papageorgiou G, Glover K, D'Mello G, Patel Y. Taking robust LPV control into flight on the VAAC Harrier. Proceedings of the 39th IEEE Conference on Decision and Control; 2000:4558-4564; Sydney, Australia.
17. Gáspár P, Szabó Z, Bokor J. A grey-box identification of an LPV vehicle model for observer-based side-slip angle estimation. Proceedings of the American Control Conf.; 2007:2961-2965; New York City.
18. Tóth R, van de Wal M, Heuberger PSC, Van den Hof PMJ. LPV identification of high performance positioning devices. Proceedings of the American Control Conference; 2011:151-158; San Francisco, CA.
19. Leith DJ, Leithhead WE. Gain-scheduled controller design: an analytic framework directly incorporating non-equilibrium plant dynamics. *Int J Control*. 1998;70:249-269.
20. Marcos A, Balas GJ. Development of linear-parameter-varying models for aircraft. *J Guid Control Dyn*. 2004;27(2):218-228.
21. Donida F, Romani C, Casella F, Lovera M. Towards integrated modeling and parameter estimation: an LFT-Modelica approach. Proceedings of the 15th IFAC Symposium on System Identification; 2009:1286-1291; Saint-Malo, France.
22. Kwiatkowski A, Werner H, Boll MT. Automated generation and assessment of affine LPV models. Proceedings of the 45th IEEE Conf. on Decision and Control; 2006:6690-6695; San Diego, CA.
23. Hoffmann C, Werner H. LFT-LPV modeling and control of a control moment gyroscope. Proceedings of the 54th IEEE Conference on Decision and Control; 2015:5328-5333; Osaka, Japan.
24. Abbas H, Tóth R, Petreczky M, Meskin N, Mohammadpour J. Embedding of nonlinear systems in a linear parameter-varying representation. Proceedings of the 19th IFAC World Congress; 2014; Cape Town, South Africa.
25. Tóth R, Willems JC, Heuberger PSC, Van den Hof PMJ. The behavioral approach to linear parameter-varying systems. *IEEE Trans Automat Contr*. 2011;56:2499-2514.
26. Tóth R, Abbas H, Werner W. On the state-space realization of LPV input-output models: practical approaches. *IEEE Trans Control Syst Technol*. 2012;20:139-153.
27. Nijmeijer H, van der Schaft AJ. *Nonlinear Dynamic Control Systems*. Springer; 1996.
28. Henson M, Seborg D. *Nonlinear Process Control*. Englewood Cliffs; 1998.
29. Teel A, Praly L. Global stabilizability and observability imply semi-global stabilizability by output feedback. *Syst Control Lett*. 1994;22(5):313-325.
30. Gauthier JP, Bornard G. Observability for any $u(t)$ of a class of nonlinear systems. *IEEE Trans Automat Contr*. 1981;26(4):922-926.
31. Cox AD, Little J, O'Shea D. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra (Undergraduate Texts in Mathematics)*. Springer; 2007.
32. Hoffmann C, Werner H. Compact LFT-LPV modeling with automated parameterization for efficient LPV controller synthesis. Proceedings of the American Control Conference; 2015:119-124; Chicago, IL.
33. Hashemi S, Gürcüoğlu U, Werner H. Interaction control of an industrial manipulator using LPV techniques. *Mechatronics*. 2013;23(6):689-699.
34. Ruderman M, Krettek J, Hoffmann F, Bertram T. Optimal state space control of DC motor. Proceedings of the 17th IFAC World Congress; 2008:5796-5801; Soul, South Korea.
35. Atkinson KE. *An Introduction to Numerical Analysis*. John Wiley and Sons; 1989.
36. Shao X, Ma C. A general approach to derivative calculation using wavelet transform. *Chemom Intell Lab Syst*. 2003;69(1):157-165.
37. Savitzky A, Golay M. Smoothing and differentiation of data by simplified least squares procedures. *Anal Chem*. 1964;36(8):1627-1639.
38. Pintelon R, Schoukens J. Real-time integration and differentiation of analog signals by means of digital filtering. *IEEE Trans Instrum Meas*. 1990;39(6):923-927.
39. Rabiner L, Steiglitz K. The design of wide-band recursive and nonrecursive digital differentiators. *IEEE Trans Audio Electroacoust*. 1970;18(2):204-209.
40. Ferrer-Arnau L, Mon-Gonzalez J, Parisi-Baradad V. Operators to calculate the derivative of digital signals. Proceedings of the 14th Workshop on Advances in Instrumentation and Sensors Interoperability; 2013:301-306; Barcelona, Spain.
41. Abbasi F, Mohammadpour J, Tóth R, Meskin N. A Bayesian approach for model identification of LPV systems with uncertain scheduling variables. Proceedings of the 54th IEEE Conference on Decision and Control; 2015:789-794; Osaka, Japan.
42. Cerone V, Regruto D. Set-membership identification of LPV models with uncertain measurements of the time-varying parameter. Proceedings of the 47th IEEE Conference on Decision and Control; 2008:4491-4496; Cancun, Mexico.
43. Sadeghzadeh A. LMI relaxations for robust gain-scheduled control of uncertain linear parameter varying systems. *IET Control Theory Appl*. 2019;13(4):486-495.
44. Dieudonne J. *Foundations of Modern Analysis, Volume 1*. Academic Press; 1969.
45. Sugie T, Shimizu K, Imura J. $\mathcal{H}_\infty$ control with exact linearization and its application to magnetic levitation systems. Proceedings of the 12th IFAC World Congress; 1993:363-366; Sydney, Australia.

46. Koelewijn PJW, Tóth R. Physical parameter estimation of an unbalanced disc system. Technical report, TUE-CS-2019; 2019. Eindhoven University of Technology.
47. Apkarian P, Gahinet P. A convex characterization of gain-scheduled $\mathcal{H}_\infty$ controllers. *IEEE Trans Automat Contr*. 1995;40(5):853-864.
48. Skogestad S. Dynamics and control of distillation columns: a tutorial introduction. *Chem Eng Res Des*. 1997;75(6):539-562.
49. Morato MM, Normey-Rico JE, Sename O. Model predictive control design for linear parameter varying systems: a survey. *Annu Rev Control*. 2020;49:64-80.
50. Kanthasamy R, Hisyam A, Aziz N, Abd Shukor SR. Nonlinear model predictive control of a distillation column using Wavenet based Hammerstein model. *Eng Lett*. 2012;20:1-6.

How to cite this article: Abbas HS, Tóth R, Petreczky M, Meskin N, Mohammadpour Velni J, Koelewijn PJ.W.. LPV modeling of nonlinear systems: A multi-path feedback linearization approach. *Int J Robust Nonlinear Control*. 2021;31(18):9436-9465. https://doi.org/10.1002/rnc.5799