Proceedings of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Embedding of Nonlinear Systems in a Linear Parameter-Varying Representation

Hossam S. Abbas [*,†] **Roland Tóth** [**] **Mihály Petreczky** [***]
Nader Meskin [*] **Javad Mohammadpour** [****]

[*] *Department of Electrical Engineering, College of Engineering, Qatar University, PO Box 2713 Doha, Qatar, (e-mail: {hossam.abbas,nader.meskin}@qu.edu.qa)*
[**] *Control Systems Group, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands, (e-mail: r.toth@tue.nl)*
[***] *Department of Computer Science and Automatic Control, Ecole des Mines de Douai, 59508 Douai, France, (e-mail: mihaly.petreczky@mines-douai.fr)*
[****] *Intelligent Control Systems Lab, College of Engineering, The University of Georgia, Athens, GA 30602, USA, (e-mail: javadm@uga.edu)*
[†] *Electrical Engineering Department, Faculty of Engineering, Assiut University, Assiut, Egypt*

Abstract: This paper introduces a systematic approach to synthesize linear parameter-varying (LPV) representations of nonlinear (NL) systems which are originally defined by control affine state-space representations. The conversion approach results in LPV state-space representations in the observable canonical form. Based on the relative degree concept of NL systems, the states of a given NL representation are transformed to new coordinates that provide its normal form. In the SISO case, all nonlinearities of the original system are embedded in one NL function which is factorized to construct the LPV form. An algorithms is proposed for this purpose. The resulting transformation yields an LPV model where the scheduling parameter depends on the derivatives of the inputs and outputs of the system. In addition, if the states of the NL model can be measured or estimated, then the procedure can be modified to provide LPV models scheduled by these states. Examples are included for illustration.

1. INTRODUCTION

The framework of *linear parameter-varying* (LPV) systems was introduced to address the control of *nonlinear* (NL) and *time-varying* (TV) systems using the extension of powerful *linear time-invariant* (LTI) approaches such as $\mathcal{H}_2/\mathcal{H}_\infty$ optimal control, see e.g., [Scherer, 1996]. LPV systems are dynamical 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, which is assumed to be measurable and free (external) in the modeled system. In this way, variation of p represents time-variance, changing operating conditions, etc., and aims at embedding the original NL/TV behavior into the solution set of an LPV system [Rugh and Shamma, 2000, Tóth, 2010]. However, in many practical systems, p is often associated with inputs, outputs or states of the modeled system (e.g., consider operating conditions), which contradicts its assumed property of being free. Such situations are often labeled to be *quasi*-LPV (q-LPV), however what really happens is that the assumed freedom of p introduces conservativeness in the embedding Hence, one important objective of LPV modeling, besides of achieving complete embedding, is to minimize such conservativeness.

Existing approaches for the LPV modeling of NL dynamical systems can be classified into three main categories: linearization-based (including multiple-model design), state-transformation & function substitution-based, and automated conversion procedures [Tóth, 2010]. In the first category, a NL description of the system is linearized at several operating points, then the resulting linearized models are interpolated to get a global approximation of the system in an LPV *state-space* (LPV-SS) form, see e.g., [Petersson and Löfberg, 2012]. State-transformation approaches, like [Shamma and Cloutier, 1993], start with a priori choice of states being p and try to apply a coordinate change of the NL-SS representation to arrive to an LPV form. Substitution-based approaches try to rewrite the NL-SS representation in a form where NL terms can be absorbed by p, see e.g., [Leith and Leithhead, 1998]. The first methodology usually provides an approximation of the system in an LPV form which is only descriptive for slow variations of the operating point, whereas the others usually produce an exact q-LPV representation; however they are applicable only for a limited class of NL representations, see [Tóth, 2010] for more details. The last category stands for the automated approaches that try to find an exact q-LPV representation with least possible conservativeness, e.g., [Donida et al., 2009, Kwiatkowski et al., 2006, Tóth, 2010]. However, they are computationally intensive algorithms and provide little system theoretic understanding of the choices taken.

In general [1], the existing techniques do not pay serious attention to several issues regarding the resulting LPV models, namely: preservation of structural properties (minimality, controllability, etc.), singularity points of the system, 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 of existing techniques are based on ad-hoc mathematical manipulations (non-unique, non-systematic, etc.) and require a serious level of experience to be used. Moreover, often the scheduling parameters of the resulting LPV model necessarily depend on the states of the original NL system, which might not be accessible in practice.

In this paper, inspired by the feedback linearization theory, a systematic procedure is proposed to convert control affine NL state-space representation into state minimal LPV-SS representations in an observable canonical form. A particular advantage of this canonical form that it can be directly converted into an equivalent LPV input-output form using recently developed LPV realization theory [Tóth, 2010]. This way the obtained form is useful both for control synthesis and model structure selection. This procedure can be seen as a novel state transformation approach as the idea is based on transforming the states of a given NL representations into the normal form such that in the SISO case all nonlinearities in the NL model are realized in only one NL term. Then, an exact substitution-based technique is presented to provide the LPV model. The state transformation leads to the systematic construction of scheduling signals which depend on the inputs, outputs, and their derivatives if the original states of the NL model cannot be provided during practical implementation. Explanation why such a scheduling construction is practically useful will be analyzed in detail. Examples are also provided to illustrate the procedure.

The paper is organized as follows: Section 2 introduces the concept of LPV representations considered in this work. The proposed NL to LPV model conversion procedure is described in Section 4. A modification of this procedure is presented in Section 4 to give an alternative conversion scheme in case of low relative degree of the system. The examples are given in Section 5 and conclusions are drawn in Section 6.

2. LPV REPRESENTATIONS

Continuous-time state-space representation of general LPV systems is defined as [Tóth, 2010]:

$$\frac{d}{dt}z = (A \diamond p)z + (B \diamond p)u, \tag{1a}$$

$$y = (C \diamond p)z + (D \diamond p)u, \tag{1b}$$

where $u : \mathbb{T} \to \mathbb{R}^{n_\mathrm{u}}$, $y : \mathbb{T} \to \mathbb{R}^{n_\mathrm{y}}$, $z : \mathbb{T} \to \mathbb{R}^n$ and $p : \mathbb{T} \to \mathbb{P}$ are the input, output, state, scheduling signals of the system, respectively, $\mathbb{T} = \mathbb{R}$ is the time axis, and $\mathbb{P} \subseteq \mathbb{R}^{n_\mathrm{p}}$ denotes the scheduling set which is assumed to be compact. Furthermore, $A, \ldots, D$ in (1) are matrices of real meromorphic functions [2] with a finite number of *essential* arguments and the operator $\diamond$ is used as a short hand notation for the evaluation of these functions on a given trajectory of p and with a unique indexing of the arguments based on the sequence $p, \frac{d}{dt}p, \frac{d^2}{dt^2}p, \ldots$, i.e.,

$$(X \diamond p) := X(p, \frac{d}{dt}p, \frac{d^2}{dt^2}p, \ldots). \tag{2}$$

In other words, this operator expresses the evaluation of the function X along a scheduling trajectory p and its derivatives, corresponding to a dynamic mapping between p and X or so called *dynamic dependence*. Dependence on the value of $p(t)$ only is called *static dependence*. For simplicity of presentation, we consider in the sequel SISO systems only, i.e., $n_\mathrm{u} = n_\mathrm{y} = 1$ in (1); furthermore we assume w.l.o.g. that $(D \diamond p) = 0$ as this term is always eliminable via y. We frequently drop the dependence on time t to simplify the notation.

Canonical forms of state-space representations are important as they provide the common ground of equivalent transformations between state-space and input-output representations. Here, we are interested in the so-called LPV observability canonical form as the proposed NL to LPV conversion procedure results in such form. Moreover, transforming an LPV-SS observable canonical form to an LPV-IO form and vice-versa has a direct formula based solution, while this is not true for other state-basis in general, see [Tóth, 2010]. The structure of the system matrices in the observability canonical representation are given by [Tóth, 2010]:

$$\left[\begin{array}{c|c} (A \diamond p) & (B \diamond p) \\ \hline (C \diamond p) & 0 \end{array}\right] = \left[\begin{array}{cccc|c} 0 & 1 & \ldots & 0 & \beta_{n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & \beta_1 \\ \alpha_0 & \alpha_1 & \ldots & \alpha_{n-1} & \beta_0 \\ \hline 1 & 0 & \ldots & 0 & 0 \end{array}\right] \diamond p, \tag{3}$$

for the SISO case, where $\{\alpha_i\}_{i=1}^{n-1}$, $\{\beta_i\}_{i=1}^{n-1}$ are meromorphic. A special form of (3) is given by considering $\beta_i = 0$, $i = 1, \ldots, n-1$, which is of particular importance in this work and will be referred in the sequel to as the *simplified* observability form whereas the form (3) will be referred to as the *full* observability form.

3. CONVERSION TO THE SIMPLIFIED OBSERVABILITY FORM

Consider a SISO NL system represented by

$$\frac{d}{dt}x = f(x) + g(x)u, \tag{4a}$$

$$y = h(x), \tag{4b}$$

where $x : \mathbb{T} \to \mathbb{X}$ is the state vector of the system, $\mathbb{X}$ is an open set of $\mathbb{R}^n$ and $f(x) : \mathbb{X} \to \mathbb{R}^n$, $g(x) : \mathbb{X} \to \mathbb{R}$, $h(x) : \mathbb{X} \to \mathbb{R}$ are assumed [3] to be real-valued analytic functions ($\mathcal{C}^\infty$ functions) of x defined on $\mathbb{X}$. This means that any order partial derivatives of these functions exist and are

[1] except the decision tree algorithm in [Tóth, 2010].

[2] $f : \mathbb{R}^k \to \mathbb{R}$ is a real meromorphic function if $f = \frac{g}{h}$, where $g, h : \mathbb{R}^k \to \mathbb{R}$ are holomorphic (analytic) and $h \neq 0$.

[3] This assumption is due to technical convenience and can be relaxed to "smooth enough" (at most $\mathcal{C}^n$) functions in terms of the proposed conversion approach.

continuous. The form (4) represents a class of NL systems commonly referred to as *control-affine* NL systems; it constitutes a class of mechanical systems which are often encountered in applications [Nijmeijer and van der Schaft, 1990]; furthermore, it can describe many first-principles models used in process systems [Henson and Seborg, 1998].

The problem investigated in this section is to convert NL systems represented by (4) with $x \in \mathbb{X}_0$ to the simplified LPV observable form, where $\mathbb{X}_0 \subseteq \mathbb{X}$ is an open neighborhood of x_0. The basic idea is to first transform the NL representation (4) into the so-called *normal* form, [Isidori, 1995], using coordinate transformation, and then apply factorization to convert the resulting normal form to the simplified LPV observability form.

To transform (4) to the normal form, one has to construct a coordinate transformation as

$$z = (\Phi(x) \diamond u) = [(\phi_1(x) \diamond u) \ \dots \ (\phi_n(x) \diamond u)]^\top, \quad (5)$$

where $(\Phi(x) \diamond u)$ denotes a smooth function defined on $\mathbb{X}_0$ with static dependence on x and dynamic dependence on u. The resulting normal form is given by

$$\frac{d}{dt}z_1 = z_2, \ \dots \ \frac{d}{dt}z_{n-1} = z_{n-2}, \ \frac{d}{dt}z_n = \bar{\alpha} \diamond (y, u), \quad (6a)$$

$$y = z_1, \quad (6b)$$

where $\bar{\alpha}$ is a smooth function with dynamic dependence on (y, u). The mapping Φ should define a local diffeomorphism, i.e., its Jacobian matrix should be nonsingular at x_0, in order to qualify as a local coordinate transformation on the open set $\mathbb{X}_0$ [Isidori, 1995]. Moreover, we restrict u, such that $(\Phi(x) \diamond u)$ is a local diffeomorphism, which imposes constraints on u and its derivatives. Now, given the representation (6), if $\bar{\alpha}$ can be factorized as

$$\bar{\alpha} \diamond (y, u) = \big(\gamma \diamond (y, u)\big) + \big(\beta_0 \diamond (y, u)\big)u + \sum_{i=0}^{n-1} \big(\alpha_i \diamond (y, u)\big) z_{i+1}, \quad (7)$$

then the simplified LPV observable canonical form (3) obtained with the additional term γ. The scheduling variable p of the resulting LPV representation includes elements of (u, y). Next, we characterize the scheduling set $\mathbb{P}$ of the resulting LPV form. Let us define the so-called *latent behavior* of the NL representation (4) with $x \in \mathbb{X}_0$ and $u \in \mathbb{D} \subseteq \mathbb{R}$ as follows:

$$\mathfrak{B}_\mathrm{L} := \big\{(y, u, x) \in \mathcal{C}^{m_\mathrm{y}}(\mathbb{T}, \mathbb{R}) \times \mathcal{C}^{m_\mathrm{u}}(\mathbb{T}, \mathbb{D}) \times \mathcal{C}^1(\mathbb{T}, \mathbb{X}_0) \text{ s.t. (6) is satisfied}\big\},$$

where $\mathcal{C}^k(\mathbb{T}, \mathbb{R})$ denotes the space of k continuously differentiable functions $\mathbb{T} \to \mathbb{R}$. The behavior $\mathfrak{B}_\mathrm{L}$ is a set of admissible solutions of (4) such that $x \in \mathbb{X}_0$ and $u \in \mathbb{D}$. In addition, we define the so-called *manifest behavior* of the NL representation (4) with $x \in \mathbb{X}_0$ and $u \in \mathbb{D}$ as follows

$$\mathfrak{B} := \big\{(y, u) \in \mathcal{C}^{m_\mathrm{y}}(\mathbb{T}, \mathbb{R}) \times \mathcal{C}^{m_\mathrm{u}}(\mathbb{T}, \mathbb{D}) \mid \exists\, x \in \mathcal{C}^1(\mathbb{T}, \mathbb{X}_0) \text{ s.t. } (y, u, x) \in \mathfrak{B}_\mathrm{L}\big\},$$

which is the set of admissible input-output trajectories of (4) such that $x \in \mathbb{X}_0$ and $u \in \mathbb{D}$; for more details see [Willems, 2007]. Then, the scheduling set $\mathbb{P}$ can be defined as

$$\mathbb{P} := \prod_{\nu=0}^{m_\mathrm{y}} \mathbb{Y}_\nu \times \prod_{\mu=0}^{m_\mathrm{u}} \mathbb{U}_\mu, \quad (8)$$

where

$$\mathbb{Y}_\nu \subseteq \left\{ \frac{d^\nu}{dt^\nu}y \mid \frac{d^i}{dt^i}y \in \mathbb{Y}_i, \text{ for } 0 \le i < \nu, \ \exists u \in \mathcal{C}^{m_\mathrm{u}}(\mathbb{T}, \mathbb{D}), \ \frac{d^j}{dt^j}u \in \mathbb{U}_j \text{ for } 0 \le j < \nu, \text{ s.t. } (y, u) \in \mathfrak{B} \right\},$$

$$\mathbb{U}_\mu \subseteq \left\{ \frac{d^\mu}{dt^\mu}u \mid \frac{d^k}{dt^k}u \in \mathbb{U}_k, \text{ for } 0 \le k < \mu, \ \exists y \in \mathcal{C}^{m_\mathrm{y}}(\mathbb{T}, \mathbb{R}), \ \frac{d^l}{dt^l}y \in \mathbb{Y}_l, \text{ for } 0 \le l < \mu, \text{ s.t. } (y, u) \in \mathfrak{B} \right\},$$

$m_\mathrm{u} < m_\mathrm{y} \le n$ and the sets $\{\mathbb{Y}_\nu\}_{\nu=0}^{m_\mathrm{y}}$ and $\{\mathbb{U}_\mu\}_{\mu=0}^{m_\mathrm{u}}$ are recursively constructed and chosen to be compact. Note that

$$[z_1 \ z_2 \ \dots \ z_n]^\top = \left[y \ \frac{d}{dt}y \ \dots \ \frac{d^{n-1}}{dt^{n-1}}y\right]^\top. \quad (9)$$

How to perform the factorization step given by (7) will be discussed later.

The notion of relative degree, r, of NL systems, see [Isidori, 1995] for its definition, has an important role in finding the appropriate coordinate transformation. In the sequel, the conversion problem is considered for two cases: when the relative degree of the NL system is equal to its order and when it is less.

3.1 Case 1: Relative degree equals system order

Consider the NL system given by (4). Assume that its relative degree is well defined at the point x_0 and equals its order, i.e., $r = n$. Then, computing y and its derivatives $\frac{d^k}{dt^k}y$, for $k = 1, \dots, n$ gives

$$\begin{aligned} &y = h(x), \quad \frac{d}{dt}y = L_f h(x), \quad \frac{d^2}{dt^2}y = L_f^2 h(x), \\ &\dots \quad \frac{d^n}{dt^n}y = L_f^n h(x) + L_g L_f^{n-1} h(x) u. \end{aligned} \quad (10)$$

where $L_X^i V(\cdot)$ stands for the i^{th} Lie derivative of V w.r.t. X (see [Isidori, 1995]). Then, a local coordinate transformation around x_0 can be defined by

$$\Phi(x) = \big[h(x) \ L_f h(x) \ \dots \ L_f^{n-1} h(x)\big]^\top. \quad (11)$$

The following results are important:

Lemma 1. (Isidori [1995]). If the relative degree r of the system (4) is n at a point x_0, then the gradients $\nabla h(x_0)$, $\nabla L_f h(x_0), \dots, \nabla L_f^{n-1} h(x_0)$ are linearly independent.

Proposition 2. (Isidori [1995]). If $\Phi(x)$ is a smooth function defined on some open subset $\mathbb{X}$ of $\mathbb{R}^n$ and its Jacobian matrix $\nabla\Phi(x)$ is nonsingular at a point $x = x_0$, then, on a suitable open set $\mathbb{X}_0$ of $\mathbb{X}$, containing x_0, $\Phi(x)$ defines a local diffeomorphism.

Lemma 1 shows that $\nabla\Phi(x_0)$ is nonsingular; then, based on Proposition 2, $\Phi(x)$ in (11) defines a local diffeomorphism on $\mathbb{X}_0$, and hence, it qualifies as a local transformation. Thus, representation (4) in the new coordinates $z_i = \phi_i(x)$, $i = 1, \dots, n$ can be obtained as

$$\frac{d}{dt}z_1 = \frac{\partial}{\partial x}\phi_1\frac{dx}{dt} = L_f h(x) = \phi_2(x) = z_2,$$
$$\vdots$$
$$\frac{d}{dt}z_{n-1} = \frac{\partial}{\partial x}\phi_{n-1}\frac{dx}{dt} = L_f^{n-1}h(x) = \phi_n(x) = z_n,$$
$$\frac{d}{dt}z_n = L_f^n h(x) + L_g L_f^{n-1}h(x)u.$$

This is in the form (6) with $\bar{\alpha}$ given by

$$\bar{\alpha}(z,u) = L_f^n h(\Phi^{-1}(z)) + L_g L_f^{n-1}h(\Phi^{-1}(z))u. \quad (12)$$

Note that all nonlinearities in (4) are absorbed in the term $\bar{\alpha}$. Then, factorization of $\bar{\alpha}$, as shown in (7), results in the simplified LPV observable representation, where p includes elements of u and, due to (9), elements of y and its derivatives. Thus, the scheduling set can be defined as shown in (8) with $m_\mathrm{u} = 0$.

Remark 3. Several NL to LPV conversion approaches in the literature provide LPV models in which the scheduling variable p includes all or part of the original states of the NL representation. These states are are often difficult or expensive (or even not possible) to be accurately measured or estimated (like in several systems in process control). In the proposed conversion method, p includes elements of u, and all or part of the new coordinates $z_1, \ldots, z_n$, which can be measured and estimated directly as they represent the system output and its derivatives. Alternatively, they can also be computed from the original states $x_1, \ldots, x_n$ of the NL system if they are available as shown in (10). This means that a good compromise can be found w.r.t. the implementability and utilization of the resulting LPV form, which is unique among the conversion techniques.

Remark 4. It is also important to highlight that the resulting LPV description will only give a representation of the original NL system around $\mathbb{X}_0$. On one hand, this gives system theoretic guarantees where the LPV representation is valid, which has paramount importance for control and identification. On the other hand, this condition can be relaxed by exploiting the meromorphic nature of the coefficient functions and establishing representation of the behavior in an *almost everywhere* sense. Due to space restrictions, this relaxation is not discussed here.

3.2 Case 2: Relative degree less than system order

In this section, the local coordinate transformation is introduced when $r < n$ to convert the NL representation (4) into a simplified LPV observable form. Assume that the relative degree of (4) is well defined at a point x_0 and less than its order, i.e., $r < n$. Then, computing $y, \frac{d}{dt}y, \ldots, \frac{d^{r-1}}{dt^{r-1}}y$ in a similar way as shown in (10) can provide part of the new state coordinates such that

$$[z_1\ z_2\ \ldots\ z_{r-1}]^\top = \left[y\ \frac{d}{dt}y\ \ldots\ \frac{d^{r-1}}{dt^{r-1}}y\right]^\top$$
$$= \left[h(x)\ L_f h(x)\ \ldots\ L_f^{r-1}h(x)\right]^\top \quad (13)$$

and

$$\frac{d}{dt}z_r = \frac{d^r}{dt^r}y = L_f^r h(x) + L_g L_f^{r-1}h(x)u. \quad (14)$$

To complete the new coordinates, let

$$\frac{d}{dt}z_r = z_{r+1}.$$

Then, compute

$$\frac{d}{dt}z_{r+1} = L_f(z_{r+1}) + L_g(z_{r+1})u + \frac{\partial}{\partial u}(z_{r+1})\frac{d}{dt}u. \quad (15)$$

Next, define z_{r+2} as the r.h.s. of (15), and then compute $\frac{d}{dt}z_{r+2}$. Repeating this operation till z_n transforms (4) to the the normal form (6). Note that again, all nonlinearities of (4) are absorbed in the function $\bar{\alpha}$. This suggests a coordinate transformation given by (5), which is a smooth function of not only x but also of u, and its derivatives till the $(n-r)$-th derivative. Therefore, to be qualified as a local diffeomorphism on $\mathbb{X}_0$, the Jacobian of Φ at the point x_0, $\nabla\Phi|_{x_0}$, should be nonsingular for specified ranges of $u, \frac{d}{dt}u, \ldots, \frac{d^{n-r}}{dt^{n-r}}u$. In this case, $\nabla\Phi|_{x_0}$ is a matrix valued function of $u, \frac{d}{dt}u, \ldots, \frac{d^{n-r}}{dt^{n-r}}u$ and the ranges of the latter variables for which $\nabla\Phi|_{x_0}$ is nonsingular should be specified. This can be tackled by using the approach proposed by [Cerone et al., 2012] to evaluate non-singularity of matrices with uncertain entries by formulating this problem as a non-convex polynomial optimization problem, whose approximate solution is efficiently computed by means of sum-of-square convex-relaxation techniques. Therefore, one can check that $\nabla\Phi|_{x_0}$ is nonsingular and obtain the corresponding ranges of $u, \frac{d}{dt}u, \ldots, \frac{d^{n-r}}{dt^{n-r}}u$, which needs to be considered for the scheduling set of the resulting LPV model (see (8)).

Finally, the simplified LPV observable form can be constructed by factorizing $\bar{\alpha}$ as shown in (7), which specifies the functions $\{\alpha_i\}_{i=0}^{n-1}$, and β_0. The resulting scheduling variable may include elements of (u, y) and their derivatives according to the functions $\{\alpha_i\}_{i=0}^{n-1}$ and β_0.

Compared to the previous case, this means that the price to pay for a low relative degree of the system and excluding the original states $x_1, x_2, \ldots, x_n$ to be chosen as scheduling variables, is that the derivatives of u up to the $(n-r)$-th order might be needed to be chosen as scheduling variables to guarantee the embedding.

3.3 Factorization Algorithm

In this section, a factorization algorithm is introduced to perform the factorization step in (7).

Note that, in implementation of controllers on modern hardware, the input signal, i.e., control output, is computed by micro controllers, therefore, it can be considered as a noise-free signal, under the assumption that all actuator nonlinearites have been incorporated in the NL plant model already. Hence, for the sake of simplicity, the factorization algorithm proposed in this section will not restrict the inclusion of derivatives of u into p.

Recall again that, as specified in (9), the output and its derivatives are the states of the obtained LPV representation. Then, to cope with the above defined objectives, Algorithm 1 is introduced for the factorization step of (7). The idea of this algorithm is to first explore the prior nonlinear structure in terms of irreducible summands, and then successively factorize out linear terms from the summands where priority is given for the higher-order derivatives of y. Note that Algorithm 1 is a simplified version of the method introduced in [Kwiatkowski et al., 2006] and its improved version in [Tóth, 2010], where the factorization problem is considered for the whole nonlinear model whereas here it is dedicated only for the function $\bar{\alpha}$ with a particular prioritization of the signals.

Algorithm 1 Factorization algorithm

Require: Write $\bar{\alpha}$ as a combination of additive summands $\sum_{i=1}^{n_{\breve{\alpha}}} \breve{\alpha}_i$, where each term $\breve{\alpha}_i$ is in the form $\breve{\alpha}_i = (\check{c}_i u^{\tau_i} \prod_{j=1}^{n} z_j^{\kappa_j})/(\psi_i)$ with $\tau_i, \kappa_j \in \mathbb{Z}_+$, $\check{c}_i$ being a non-factorizable term and ψ_i being co-prime with the numerator.

$\bar{\alpha}^{\text{res}} \leftarrow 0$, $\alpha_{j\leftarrow 0:n-1} \leftarrow 0$, $\beta_0 \leftarrow 0$
for $i \leftarrow 1 : n_{\breve{\alpha}}$ **do**
  if $\tau_i \neq 0$ **then**
    $\beta_0 \leftarrow \beta_0 + \frac{\breve{\alpha}_i}{u}$
  else
    for $j \leftarrow n : 1$ **do**
      if $\kappa_j \neq 0$ & $\breve{\alpha}_i \neq 0$ **then**
        $\alpha_i \leftarrow \alpha_i + \frac{\breve{\alpha}_i}{x_j}$
        $\breve{\alpha}_i \leftarrow 0$
      end if
    end for
    if $\breve{\alpha}_i \neq 0$ **then**
      $\bar{\alpha}^{\text{res}} \leftarrow \bar{\alpha}^{\text{res}} + \breve{\alpha}_i$
    end if
  end if
end for
return $\bar{\alpha}^{\text{res}}, \alpha_0, \ldots, \alpha_{n-1}, \beta_0$

The factorization algorithms presented here might return $\bar{\alpha}^{\text{res}} \neq 0$. In that case, $\gamma = \bar{\alpha}^{\text{res}}$ in (7). Next, we suggest some approaches to deal with such term if the LPV model is to be used for control design purposes. As the simplified LPV observable form is considered here, (1a) of the LPV model with a non-factorizable term can be seen as

$$\frac{d}{dt}\begin{bmatrix} z_1 \\ \vdots \\ z_{n-1} \\ z_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \\ \alpha_0 & \alpha_1 & \ldots & \alpha_{n-1} \end{bmatrix} \begin{bmatrix} z_1 \\ \vdots \\ z_{n-1} \\ z_n \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ \vdots \\ 0 \\ \beta_0 \end{bmatrix}}_{B} u + \underbrace{\begin{bmatrix} 0 \\ \vdots \\ 0 \\ \gamma \end{bmatrix}}_{E}. \tag{16}$$

From (16), one can observe that the vector E is spanned by the vector B, and consequently, the non-factorizable term can be seen as an input disturbance to the system. Therefore, it can be treated by one of the following ways.

(1) It can be completely ignored during the control design phase, and hence its effect can be reduced by one of the following methods:
  (i) Input disturbance rejection can be considered as one of the control design objectives.
  (ii) The designed controller can be augmented with a feedforward path to compensate this non-factorizable term during control implementation, see, e.g., [Hashemi et al., 2013].
(2) One can introduce a new input w to the system to include the non-factorizable term as $w = \beta_0 u + \gamma$, which leads to $\beta_0 = 1$ in (16).
(3) It can be rewritten as $\frac{\gamma}{u}u$ or $\frac{\gamma}{z_j}z_j$, hence it can be assigned to u or to j^{th} state, respectively, as a coefficient which can be added to β_0 or α_j, respectively. This assignment gives $n+1$ possibilities depending on which component of z is used. It should be taken into consideration that u or the state z_j should not approach to zero during operation, see [Kwiatkowski et al., 2006],[Tóth, 2010] for more details.
(4) One can approximate the non-factorizable term in terms of its truncated series expansion in its arguments to obtain a completely factorizable form.

4. CONVERSION TO THE FULL OBSERVABILITY FORM

Based on the same idea presented above, next, a procedure is presented to transform a control-affine NL systems (4) to a full LPV observable form. This alternative is of particular interest for systems with low relative degree. The resulting LPV model is associated with scheduling signals that do not include input derivatives; however they depend on the availability of the original states of the system (4), and here it is assumed that they can be estimated or measured as in several mechanical systems.

Consider again the NL representation (4) with a well defined relative degree $r < n$ at a point $x = x_0$. Based upon the formulation given by (13) and (14), one can compute the new coordinates as follows. Let $z_{r+1} = L_f^r h(x)$, then

$$\frac{d}{dt} z_{r+1} = L_f^{r+1} h(x) + L_g L_f^r h(x) u.$$

Next taking $z_{r+2} = L_f^{r+1} h(x)$ provides

$$\frac{d}{dt} z_{r+2} = L_f^{r+2} h(x) + L_g L_f^{r+1} h(x) u.$$

Repeating this operation $n - r + 2$ times gives

$$\frac{d}{dt} z_1 = z_2, \tag{17a}$$
$$\vdots$$
$$\frac{d}{dt} z_r = z_{r+1} + L_g L_f^{r-1} h(x) u, \tag{17b}$$
$$\vdots$$
$$\frac{d}{dt} z_n = L_f^n h(x) + L_g L_f^{n-1} h(x) u. \tag{17c}$$

Based on this reformulation, the local coordinate transformation $\Phi(x)$ that converts (4) into (17) is given in the form of (11) again. Now it is required to show that $\Phi(x)$ is a local diffeomorphism in order to prove that (17) is an equivalent representation in the neighborhood x_0. For that, it suffices to show that $\nabla\Phi(x)|_{x_0}$ is invertible. Lemma 1 guarantees that the gradients of the first r components of $\Phi(x)$ are linearly independent (if the relative degree of the system (4) is r at $x = x_0$). If the remaining $n - r$ components are also linearly independent, then based on Proposition 2, $\Phi(x)$ defines a local diffeomorphism and the vector z qualifies as a state variable. Therefore, we can define $x = \Phi^{-1}(z)$ and substitute this in (17) so that $\bar{\alpha}(z, u)$ can be given by (12). Now, the full LPV observable form can be constructed similarly as before, except the extra terms $\{\beta_j\}_{j=1}^{n-r}$, which are assigned as

$$\beta_j = L_g L_f^{n-j-1} h(\Phi^{-1}(z)). \tag{18}$$

After factorization of the $\bar{\alpha}$ term by Algorithm 1. The scheduling variable of the resulting LPV representation may depend on the elements of the new coordinates $z_1, \ldots, z_n$ and possibly the system input. Note that the first $r - 1$ terms of the the new coordinates are the output and its $1, \ldots, r-1$-derivatives while the rest should be computed from $L_f^r h(x), \ldots, L_f^n h(x)$ given the original states $x_1, x_2, \ldots, x_n$ of the NL system. In contrast with the procedure introduced in Section 3.2, the scheduling signals here do not include the derivatives of the input and the representation is not in the normal form.

5. NUMERICAL EXAMPLES

5.1 Example 1

Consider the NL system (4) with

$$f(x) = \begin{bmatrix} 0 \\ x_1 + x_3^2 \\ x_2 + x_2 x_3 \end{bmatrix}, \; g(x) = \begin{bmatrix} x_2^2 + x_3 \\ 0 \\ 0 \end{bmatrix}, \; h(x) = x_3. \tag{19}$$

Next, this NL representation is converted into the simplified LPV observable form. The system has a relative degree 3, i.e., $r = n$, at each point $x_0 \in \mathbb{V} = \{x \in \mathbb{R}^3 \mid (x_2^2 + x_3)(x_3+1) \neq 0\}$. Therefore, we can provide a normal form of the system for any open set $X_0 \subset \mathbb{V}$. According to (11), the local coordinate transformation is

$$z = \Phi(x) = \left[x_3 \;\; x_2 + x_2 x_3 \;\; (x_3+1)(x_2^2 + x_3^2 + x_1)\right]^\top.$$

The Jacobian matrix of $\Phi(x)$ is singular at $\mathbb{R}^3 \setminus \mathbb{V}$. The inverse transformation $\Phi^{-1}(z)$ can be given by

$$x = \left[\frac{\left(z_3 - (z_1+1)^3\left(z_2^2 + z_1^2(z_1+1)^2\right)\right)}{(z_1+1)} \;\; \frac{z_2}{z_1+1} \;\; z_1\right]^\top.$$

As Φ transforms the NL model to the normal form given by (6), the function $\bar{\alpha}$ is directly obtained. The next step is to factorize $\bar{\alpha}(z)$ to get the LPV model. This is accomplished by Algorithm 1, which gives

$$\alpha_0 = 2z_2(z_1+1), \qquad \alpha_1 = -\frac{2z_2^2}{(z_1+1)^2}, \qquad \alpha_2 = \frac{3z_2}{z_1+1},$$
$$\beta_0 = (z_1+1)\left(z_1 + \frac{z_2^2}{(z_1+1)^2}\right).$$

It is worth to mention that for this system, the scheduling signal is $p = [z_1 \;\; z_2]^\top$, where $z_1 = y$ and $z_2 = \frac{d}{dt}y$.

Due to $\mathbb{V}$, the region $\mathbb{P}$ for p should be chosen such that $p_1 \neq -1$ and if $p_1 < 0$ then $p_2 \neq \sqrt{-p_1}(1+p_1)$. Note that the original system is neither controllable nor observable at these points, i.e., at $\mathbb{R}^3 \setminus \mathbb{V}$. Several techniques introduced in the literature to convert an NL model into an LPV form ignore this restriction.

5.2 Example 2

In order to demonstrate the procedures presented in Sections 3.2 and 4 we now consider an NL system (4) with

$$f(x) = \begin{bmatrix} x_2 - 2x_2x_3 + x_3^2 \\ x_3 \\ \sin(x_1) \end{bmatrix}, \; g(x) = \begin{bmatrix} 4x_2x_3 \\ -2x_3 \\ 0 \end{bmatrix}, \; h(x) = x_3. \tag{20}$$

The system has a relative degree $r = 2$ which is less than its order $n = 3$. Therefore, the NL model can be converted to both the full and the simplified LPV observability forms (3). First, the former one is considered. In this case, the new coordinate z and its inverse can be computed, respectively, by

$$z = \left[x_3 \;\; \sin(x_1) \;\; \cos(x_1)(x_3^2 - 2x_2x_3 + x_2)\right]^\top,$$
$$x = \left[\sin^{-1}(z_2) \;\; \frac{-z_3 + z_1^2\sqrt{1-z_2^2}}{(2z_1-1)\sqrt{1-z_2^2}} \;\; z_1\right]^\top.$$

The new coordinates can be used to transform the NL model into the form (17) with $r = 2$, $n = 3$, then the factorization step is performed for $\bar{\alpha}$ to construct the full LPV observable form as shown in (3), where

$$\alpha_0 = \frac{2z_2z_3^2 + \left(4z_1^2 - 4z_1 + 2z_2 + 1 - 2z_1z_2\right)\sqrt{(1-z_2^2)^3}}{(2z_1-1)(z_2^2-1)},$$
$$\alpha_1 = \frac{-z_3(2 + z_3 - 2z_2^2)}{(2z_1-1)(z_2^2-1)}, \quad \alpha_3(z) = 0,$$
$$\beta_0 = \frac{-2z_1\left(2z_2z_3^2 - 2z_1^2z_2z_3\sqrt{1-z_2^2}\right)}{(2z_1-1)(z_2^2-1)} - \frac{2z_1(4z_1^2 - 4z_1 + 1)\sqrt{(1-z_2^2)^3}}{(2z_1-1)(z_2^2-1)},$$
$$\beta_1 = \frac{-4z_1\left(z_3 - z_1^2\sqrt{1-z_2^2}\right)}{2z_1-1}.$$

The transformation is not valid for $(2z_1-1)(z_2^2-1) = 0$ which corresponds to $(1-2x_3)\cos(x_1) = 0$; this should be considered when the operating region of the intended LPV model is defined via $\mathbb{P}$. Note that the original system is not controllable at theses points as well.

Now, we convert the NL model to the simplified LPV observable form. Again, the new coordinate z and its inverse are determined and given, respectively, by

$$z = \left[x_3 \;\; \sin(x_1) \;\; \cos(x_1)(x_3^2 - 2x_2x_3 + x_2 + 4x_2x_3u)\right]^\top,$$
$$x = \left[\sin^{-1}(z_2) \;\; \frac{-z_3 + z_1^2\sqrt{1-z_2^2}}{(2z_1 - 4z_1u - 1)\sqrt{1-z_2^2}} \;\; z_1\right]^\top.$$

Note that the new coordinate z and its inverse depend on the input signal u. Then the NL model can be written in the normal form (6) and the factorization step is performed for $\bar{\alpha}$ above to get the simplified LPV observable form:

$$\alpha_0 = \frac{-2z_2z_3^2 + 4\left(\frac{d}{dt}u\right)(z_2^2z_3 - z_3)}{(z_2^2-1)(4uz_1 - 2z_1 + 1)}, + \frac{\left(-1 + 4z_1 - 2z_2 - 4z_1^2 + 2z_1z_2 + 4\left(\frac{d}{dt}u\right)z_1^2\right)\sqrt{(1-z_2^2)^3}}{(z_2^2-1)(4uz_1 - 2z_1 + 1)}$$
$$\alpha_1 = \frac{z_3(2 + z_3 - 2z_2^2)}{(z_2^2-1)(4uz_1 - 2z_1 + 1)}, \qquad \alpha_2 = 0,$$
$$\beta_0 = \frac{4z_2z_3^2z_1 + 4z_2z_3(z_2^2-1) + (2 + 24z_1^2)z_1\sqrt{(1-z_2^2)^3}}{(z_2^2-1)(4uz_1 - 2z_1 + 1)} + \frac{(16uz_1 - 4z_1z_2 - 16z_1 - 48uz_1^2 + 32u^2z_1^2)z_1\sqrt{(1-z_2^2)^3}}{(z_2^2-1)(4uz_1 - 2z_1 + 1)}.$$

In this case, the transformation is not valid for $(z_2^2 - 1)(4uz_1 - 2z_1 + 1) = 0$. Here, the scheduling signals include, in addition to the new states z_1, z_2, z_3 (which are the output and its first and second derivatives) the system input and its first derivative. This is the price to pay for having the LPV representation completely independent of the original NL states. Furthermore, the new LPV representation can be easily and directly converted into an LPV-IO form.

6. CONCLUSION

In this paper, a systematic approach has been proposed to determine state-minimal LPV-SS representations in observable canonical form for control-affine NL models.

The concept of relative degree of NL systems has been used to obtain coordinate transformations that transforms the NL representation to the normal form that has only one NL term,. This single term can be factorized to arrive at a LPV representation of the NL model in a simplified observable form. A practical algorithm is introduced to perform the factorization step. The scheduling signals of the resulting LPV model can be computed from the system output, input and their derivatives. Moreover, a procedure has been introduced that can convert a NL model with a relative degree less than the order of the system to an LPV representation in the full observable canonical form, where the scheduling signals can be computed from the original states of the NL representation. This procedure can be used if these states can be measured or estimated.

ACKNOWLEDGEMENTS

This publication was made possible by the NPRP grant (No. 5-574-2-233) from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.

REFERENCES

V. Cerone, D. Piga, and D. Regruto. Robust pole placement for plants with semialgebraic parametric uncertainty. In *Proceedings of the American Control Conference*, pages 5240–5245, Montréal, Canada, 2012.

F. Donida, C. Romani, F. Casella, and M. Lovera. Towards integrated modeling and parameter estimation: an LFT-Modelica approach. In *Proc. of the 15th IFAC Symposium on System Identification*, pages 1286–1291, Saint-Malo, France, July 2009.

S. Hashemi, U. Gürcüoğlu, and H. Werner. Interaction control of an industrial manipulator using LPV techniques. *Mechatronics*, 23(6):689 – 699, 2013.

M. Henson and D. Seborg. *Nonlinear Process control*. Prentice Hall, Englewood Cliffs, New Jesrey, 1998.

A. Isidori. *Nonlinear Control Systems: An introduction*. Springer Verlag Berlin, 1995.

A. Kwiatkowski, M. Boll, and H. Werner. Automated generation and assessment of affine LPV models. In *Proc. 45th IEEE Conference on Decision and Control*, pages 6690–6695, San Diego, CA, USA, 2006.

D. J. Leith and W. E. Leithhead. Gain-scheduled nonlinear systems: Dynamic analysis by velocity based linearization families. *Int. Journal of Control*, 70:289–317, 1998.

H. Nijmeijer and A. van der Schaft. *Nonlinear dynamical control systems*. Springer-Verlag, New York, Berlin, Heidelberg, 1990.

D. Petersson and J. Löfberg. Identification of LPV state-space models using $\mathcal{H}_2$ minimisation. In A. Varga, A. Hansson, and G. Puyou, editors, *Optimization Based Clearance of Flight Control Laws*, chapter 6, pages 111–128. 2012.

W. J. Rugh and J. S. Shamma. Research on gain scheduling. *Automatica*, 36:1401–1425, 2000.

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

J. Shamma and J. Cloutier. Gain-scheduling missile autopilot design using linear parameter varying transformations. *Journ. of Guidance Control and Dyn.*, 16(2): 256–263, 1993.

R. Tóth. *Modeling and Identification of Linear Parameter-Varying Systems*. Springer–Verlag, Berlin, Heidelberg, 2010.

J. Willems. The behavioral approach to open and interconnected systems. *IEEE Control Systems Magazine*, 27(6):46 – 99, 2007.