



**HAL**  
open science

# LPV-ARX representations of LPV state-space models with affine dependence

Mihály Petreczky, Roland Tóth, Guillaume Mercère

► **To cite this version:**

Mihály Petreczky, Roland Tóth, Guillaume Mercère. LPV-ARX representations of LPV state-space models with affine dependence. *Systems and Control Letters*, 2023, 173, pp.105459. 10.1016/j.sysconle.2023.105459 . hal-03969148

**HAL Id: hal-03969148**

**<https://hal.science/hal-03969148>**

Submitted on 2 Feb 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# LPV-ARX Representations of LPV State-Space Models with Affine Dependence

Mihály Petreczky<sup>a</sup>, Roland Tóth<sup>b</sup>, Guillaume Mercère<sup>c</sup>

<sup>a</sup>*Mihály Petreczky (Corresponding author) is with Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRISTAL, F-59000 Lille, France.*

<sup>b</sup>*R. Tóth is with the Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands and the Systems and Control Laboratory, Institute for Computer Science and Control, Budapest, Hungary.*

<sup>c</sup>*Guillaume Mercère is with the University of Poitiers, Laboratoire d'Informatique et d'Automatique pour les Systèmes, Poitiers Cedex, France.*

---

## Abstract

In this paper, we show that an input-output map can be realized by a *linear parameter-varying* (LPV) state-space representation with an affine and static dependence on the scheduling variables, if and only if this input-output map satisfies certain LPV autoregressive input-output equations. The latter class of equations is linear in the derivatives (for continuous-time) or time-shifts (for discrete-time) of the outputs and control inputs, while the coefficients of this linear equation are polynomials of the shifts of the scheduling variable in discrete-time, or of the high-order derivatives of the scheduling variable in continuous-time. This result is a generalization of the well-known equivalence between linear state-space representations and autoregressive input-output models. Moreover, this result extends the results of [1] on LPV state-space representations with a dynamic and meromorphic dependence on the scheduling variables to LPV state-space representations with a static and affine dependence on the scheduling variables.

---

## 1. Introduction

*Linear parameter-varying* (LPV) systems are usually defined as linear time-varying systems, where the time varying coefficients are functions of a certain time-varying signal, the so-called *scheduling variable* [1],[2]. Practical use of LPV systems is stimulated by the fact that LPV control design [3–9] and identification [10–20] are well developed. Despite these advances, there are important gaps in system theory for LPV systems.

---

*Email addresses:* [mihaly.petreczky@centralelille.fr](mailto:mihaly.petreczky@centralelille.fr) (Mihály Petreczky),  
[r.toth@tue.nl](mailto:r.toth@tue.nl) (Roland Tóth), [guillaume.mercere@univ-poitiers.fr](mailto:guillaume.mercere@univ-poitiers.fr) (Guillaume Mercère)

One such a gap is the relationship between input-output representations and state-space representations for LPV systems. This question was addressed by [1, 2], but [1, 2] considered LPV systems in terms of both state-space and input-output representations with *nonlinear and dynamic dependence* on the scheduling variable. More precisely, the system parameters were meromorphic functions of the scheduling variables and its derivatives (continuous-times), or of the current and future values of the scheduling variable (discrete-time). However, from a practical point of view, LPV state-space representations with *static and affine dependence* (affine dependence on the instantaneous value of the scheduling variable) are preferable [9–20]. We will use the abbreviation LPV-SSA for the latter class of state-space representations.

The motivation for the contribution of this paper is as follows: the transformation from [1, 2] may yield an input-output representation with a dynamic nonlinear dependence, even if it is applied to an LPV-SSA representation. Conversely, even if we know that an input-output representation arises from an LPV-SSA representation, the transformation of [1, 2] needs not yield an LPV-SSA representation. That is, [1, 2] does not tell us which class of input-output representations corresponds exactly to the class of LPV-SSA representations. Answering this question is especially useful for system identification. Indeed, in general, identification of input-output representations is easier than that of state-space representations. However, for control design LPV-SSA representations are highly preferred. Hence, for the identification of input-output representations to be meaningful, we need to know which input-output representations correspond to LPV-SSA representations, and how to transform an identified input-output representation to an LPV-SSA representation.

The current paper is an attempt to close this gap. We consider both the *discrete-time* (DT) and the *continuous-time* (CT) cases. We show that an input-output map can be realized by a LPV-SSA representation, if and only if the input-output map satisfies a so called *linear parameter-varying autoregressive* (LPV-ARX) representation. The latter means that any input, output and scheduling signal which are consistent with the input-output map satisfy an equation, which is linear in the derivatives (CT) or time-shifts (DT) of the inputs and outputs. Moreover, the coefficients with which the derivatives or time-shifts of the outputs or inputs are multiplied are functions of the scheduling signal. More precisely, these coefficients are polynomials of the shifted scheduling variables (DT) or of high-order derivatives of the scheduling variables (CT).

For the DT case, the result was already announced in [21], but without proofs. The technical report [22] sketches some of the proofs for the DT case. For the CT case, which is much more challenging than the DT case, the result of this paper are completely new. For the proof of our results, we use [23–26], where the correspondence between bilinear state-space representations and polynomial input-output equations was shown. In contrast to [23–26], we deal with LPV models. Hence, the adaptation of the results of [23–26] to LPV models is not trivial. In [27] a characterization is presented of those LPV input-output representation which correspond to LPV state-space representations with a *nonlinear* (hence not necessarily affine) and static dependence. For this reason,

the results of [27] do not imply the results of this paper.

As it was mentioned above, [1] addressed the equivalence between a class of LPV state-space representations. However, in [1] the matrices of the LPV state-space representation were assumed to be meromorphic functions of the scheduling variable and its derivatives (in CT) or time-shifts (in DT). Similarly, in [1] the coefficients of input-output equations were assumed to be meromorphic functions of the scheduling variable and its derivatives (in CT) or time-shifts (in DT). In contrast, in this paper we consider only LPV state-space representations with matrices which are affine functions of the scheduling variable, and they do not depend on the derivatives or time-shifts of the scheduling variable. In addition, in this paper we consider input-output equations with coefficients which are polynomial functions of the scheduling variables and their derivatives (in CT) or time-shifts (in DT). The results of [1] only tell us that if an input-output behavior can be realized by an LPV-SSA, then it will satisfy a input-output equation with coefficients that can depend dynamically and meromorphically on the scheduling variables. However, [1] does not imply that these coefficients can be restricted to be only polynomials. More importantly, [1] does not characterize those input-output equations which have an LPV-SSA realisation. That is, [1] does not answer the question which input-output equations should an input-output behavior satisfy in order to be realizable by an LPV-SSA, which is useful for further utilization, such as control design.

The paper is organized as follows. In Section 2, basic notions and concepts are introduced, which are followed by the definition of LPV-SSA representations, input-output maps and LPV-ARX representations. In Section 3, the main results of the paper are stated. In Section 4, we present the proof of the main results.

## 2. Preliminaries

*Notation.* Let  $\mathbb{Z}$  and  $\mathbb{R}$  be the set of integer and real numbers, respectively, while  $\mathbb{N} = \mathbb{Z}_0^+$  and  $\mathbb{R}_0^+$  stand for the set of non-negative integers and non-negative reals. For a set  $X$ , denote by  $\mathcal{S}(X)$  the set of finite sequences generated from  $X$ , *i.e.*, each  $s \in \mathcal{S}(X)$  is of the form  $s = \zeta_1 \zeta_2 \cdots \zeta_k$  with  $\zeta_1, \zeta_2, \dots, \zeta_k \in X$ ,  $k \in \mathbb{N}$ .  $|s|$  denotes the length of the sequence  $s$ , while for  $s, r \in \mathcal{S}(X)$ ,  $sr \in \mathcal{S}(X)$  corresponds the concatenation operation. The symbol  $\varepsilon$  is used for the empty sequence and  $|\varepsilon| = 0$  with  $s\varepsilon = \varepsilon s = s$ . Furthermore,  $X^{\mathbb{N}}$  denotes the set of all functions of the form  $f : \mathbb{N} \rightarrow X$ . For each  $j = 1, \dots, m$ ,  $e_j$  is the  $j^{\text{th}}$  *standard basis* in  $\mathbb{R}^m$ . Furthermore, let  $\mathbb{I}_{s_1}^{s_2} = \{s \in \mathbb{Z} \mid s_1 \leq s \leq s_2\}$  be an index set for  $s_1, s_2 \in \mathbb{Z}$ .

Let  $\mathbb{T} = \mathbb{R}_0^+ = [0, +\infty)$  be the time axis in the CT case, and  $\mathbb{T} = \mathbb{N}$  in the DT case. Denote by  $\xi$  the differentiation operator  $\frac{d}{dt}$  (in CT) and the forward time-shift operator  $q$  (in DT), *i.e.*, if  $z : \mathbb{S} \rightarrow \mathbb{R}^n$ ,  $\mathbb{S} \subseteq \mathbb{T}$  then  $\xi z$  is a function on  $\mathbb{S}$ , such that for  $\mathbb{T} = \mathbb{R}_0^+$ ,  $(\xi z)(t) = \frac{d}{dt} z(t)$  for all  $t \in \mathbb{S}$ , and for  $\mathbb{T} = \mathbb{N}$ ,  $(\xi z)(t) = z(t+1)$  for all  $t \in \mathbb{S}$ . Note that in DT, we will apply  $\xi$  only when for every  $t \in \mathbb{S}$ ,  $t+1 \in \mathbb{S}$ . As usual, denote by  $\xi^k$  the  $k$ -fold application of  $\xi$ , *i.e.* for any  $z : \mathbb{T} \rightarrow \mathbb{R}^n$ ,  $\xi^0 z = z$ , and  $\xi^{k+1} z = \xi(\xi^k z)$  for all  $k \in \mathbb{N}$ .

A function  $f : \mathbb{R}_0^+ \rightarrow \mathbb{R}^n$  is called *piecewise-continuous*, if  $f$  has finitely many points of discontinuity on any compact subinterval of  $\mathbb{R}_0^+$  and, at any point of discontinuity, the left-hand and right-hand side limits of  $f$  exist and are finite. We denote by  $\mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R}^n)$  the set of all *piecewise-continuous functions* of the above form. We denote by  $\mathcal{C}_a(\mathbb{R}_0^+, \mathbb{R}^n)$  the set of all  $n$ -dimensional *absolutely continuous functions* [28].

*LPV-SSA representations.* Below we follow the presentation of [29]. An LPV *state-space* (SS) representation with *affine* linear dependence on the *scheduling variable* (abbreviated by LPV-SSA) is a system of the form

$$\Sigma \begin{cases} \xi x(t) &= A(p(t))x(t) + B(p(t))u(t), \\ y(t) &= C(p(t))x(t) + D(p(t))u(t), \end{cases} \quad (1)$$

where  $x(t) \in \mathbb{X} = \mathbb{R}^{n_x}$  is the state variable,  $y(t) \in \mathbb{Y} = \mathbb{R}^{n_y}$  is the (measured) output,  $u(t) \in \mathbb{U} = \mathbb{R}^{n_u}$  represents the input signal and  $p(t) \in \mathbb{P} \subseteq \mathbb{R}^{n_p}$  is the so called *scheduling variable* of the system represented by  $\Sigma$ , where the matrix functions  $A(\cdot), \dots, D(\cdot)$  defining the SS representation (1) are considered to be affine and static functions of  $p(t)$  in the form

$$\begin{aligned} A(p(t)) &= A_0 + \sum_{i=1}^{n_p} A_i p_i(t), & B(p(t)) &= B_0 + \sum_{i=1}^{n_p} B_i p_i(t), \\ C(p(t)) &= C_0 + \sum_{i=1}^{n_p} C_i p_i(t), & D(p(t)) &= D_0 + \sum_{i=1}^{n_p} D_i p_i(t), \end{aligned} \quad (2)$$

for every  $p(t) = [ p_1(t) \ \cdots \ p_{n_p}(t) ]^\top \in \mathbb{P}$ , with constant matrices  $A_i \in \mathbb{R}^{n_x \times n_x}$ ,  $B_i \in \mathbb{R}^{n_x \times n_u}$ ,  $C_i \in \mathbb{R}^{n_y \times n_x}$  and  $D_i \in \mathbb{R}^{n_y \times n_u}$  for all  $i \in \mathbb{I}_0^{n_p}$ . The signal  $p$  corresponds to varying-operating conditions, nonlinear/time-varying dynamical aspects and /or external effects influencing the plant behavior and it is allowed to vary in the set  $\mathbb{P}$ , see [1] for details.

**Remark 1.** *In the sequel, for the sake of simplicity, we assume that  $\mathbb{P} = \mathbb{R}^{n_p}$ .*

By a solution of  $\Sigma$  we mean a tuple of trajectories  $(x, y, u, p) \in (\mathcal{X}, \mathcal{Y}, \mathcal{U}, \mathcal{P})$  satisfying (1), where

- in CT,  $\mathcal{X} = \mathcal{C}_a(\mathbb{R}_0^+, \mathbb{X}), \mathcal{Y} = \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{Y}), \mathcal{U} = \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{U}), \mathcal{P} = \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{P})$ ,
- in DT  $\mathcal{X} = \mathbb{X}^{\mathbb{N}}, \mathcal{Y} = \mathbb{Y}^{\mathbb{N}}, \mathcal{U} = \mathbb{U}^{\mathbb{N}}, \mathcal{P} = \mathbb{P}^{\mathbb{N}}$ .

Note that in CT,  $(x, u, y, p)$  is assumed to satisfy (1) for almost all  $t \in \mathbb{R}_+$ .

Note that for any input and scheduling signal  $(u, p) \in \mathcal{U} \times \mathcal{P}$  and any initial state  $x_o$  and any initial time  $t_o$ , there exists a *unique* pair  $(y, x) \in \mathcal{Y} \times \mathcal{X}$  such that  $(x, y, u, p)$  is a solution of (1) and  $x(t_o) = x_o$ . That is, the dynamics of  $\Sigma$  are thus driven by the inputs  $u \in \mathcal{U}$  as well as the scheduling variables  $p \in \mathcal{P}$ .

**Remark 2** (Zero initial time). *Notice that without loss of generality, we can take initial time  $t_o = 0$ , see [2].*

Let  $x_o \in \mathbb{R}^{n_x}$  be an initial state of  $\Sigma$ . Then the *input-to-output function* of  $\Sigma$  induced by  $x_o$  is the function

$$\mathfrak{Y}_{\Sigma, x_o} : \mathcal{U} \times \mathcal{P} \rightarrow \mathcal{Y},$$

such that for any  $(u, p, y) \in \mathcal{U} \times \mathcal{P} \times \mathcal{Y}$ ,

$$y = \mathfrak{Y}_{\Sigma, x_o}(u, p)$$

if and only if there exists a state trajectory  $x \in \mathcal{X}$  such that  $(x, y, u, p)$  is a solution of (1) and  $x(0) = x_o$ .

In other words,  $\mathfrak{Y}_{\Sigma, x_o}(u, p)$  is the output trajectory of  $\Sigma$  which corresponds to the input  $u$ , scheduling signal  $p$  and a state trajectory  $x$  of  $\Sigma$  such that  $x(0) = x_o$ .

We formalize potential input-output behavior of LPV-SSA representations as functions of the form

$$\mathfrak{Y} : \mathcal{U} \times \mathcal{P} \rightarrow \mathcal{Y}. \quad (3)$$

The LPV-SSA representation  $\Sigma$  is called a *realization of the function  $\mathfrak{Y}$  of the form (3) from the initial state  $x_o$* , if  $\mathfrak{Y} = \mathfrak{Y}_{\Sigma, x_o}$ . If the initial state is not relevant, then we will say that the LPV-SSA  $\Sigma$  is a *realization of  $\mathfrak{Y}$* , if there exist an initial state  $x_o$  of  $\Sigma$ , such that  $\Sigma$  is a realization of  $\mathfrak{Y}$  from  $x_o$ . We say that  $\mathfrak{Y}$  has an LPV-SSA realization, if there exists an LPV-SSA which is a realization of  $\mathfrak{Y}$ . We refer the reader to [29] for necessary and sufficient conditions for existence of an LPV-SSA which is a realization of an input-output map  $\mathfrak{Y}$ .

Note that we can assume that  $D(\cdot) \equiv 0$  without any loss of generality regarding the concepts of realizability. Therefore, in the sequel, we will assume that  $D_i = 0$  for all  $i \in \mathbb{I}_0^{n_p}$ , and we will often use the shorthand notation

$$\Sigma = (\mathbb{P}, \{A_i, B_i, C_i\}_{i=0}^{n_p})$$

to denote an LPV-SSA representation of the form (1).

*Impulse response representation (IRR)*. Next, we recall from [21, 22, 29] the notion of an *impulse response representation* (IRR) of an input-output map both in CT and DT. Note that in [29] (in CT) and in [21, 22] (in DT), it was shown that *all input-output maps which are realizable as an LPV-SSA admit such a representation*, so for the purposes of this paper we will restrict attention to input-output maps which admit an IRR.

In order to define IRRs, we need the following notation and terminology.

**Notation 1.** Let  $p_q$  denote the  $q^{\text{th}}$  entry of the vector  $p \in \mathbb{R}^{n_p}$  if  $q \in \mathbb{I}_1^{n_p}$  and let  $p_0 = 1$ . That is, for any  $p \in \mathcal{P}$ ,  $p_0(t) = 1$  and  $p_q(t)$  is the  $q^{\text{th}}$  entry of  $p(t)$ .

**Definition 1.** For any sequence  $s \in \mathcal{S}(\mathbb{I}_0^{n_p})$ , time moments  $t, \tau \in \mathbb{T}$ ,  $\tau \leq t$ , and any scheduling trajectories  $p \in \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R}^{n_p})$  (in CT) or  $p \in (\mathbb{R}^{n_p})^{\mathbb{N}}$  (in DT), the so-called sub-Markov dependence  $(w_s \diamond p)(t, \tau)$  is defined as follows:

- **Continuous-time:** If  $s = \epsilon$  is the empty sequence, then let  $(w_\epsilon \diamond p)(t, \tau) = 1$ . If  $s = s_1 s_2 \cdots s_n$  for some  $n > 0$ ,  $s_1, \dots, s_n \in \mathbb{I}_0^{n_p}$ , then

$$(w_s \diamond p)(t, \tau) = \int_\tau^t p_{s_n}(\delta) \cdot (w_{s_1 s_2 \cdots s_{n-1}} \diamond p)(\delta, \tau) d\delta.$$

- **Discrete-time:** If  $s = \epsilon$ , then  $(w_\epsilon \diamond p)(\tau - 1, \tau) = 1$ , if  $s$  is of the form  $s = s_1 s_2 \cdots s_n$ , for  $n = t - \tau + 1$  and for some  $s_1, \dots, s_n \in \mathbb{I}_0^{n_p}$ , then

$$(w_s \diamond p)(t, \tau) = p_{s_1}(\tau) p_{s_2}(\tau + 1) \cdots p_{s_n}(t),$$

or otherwise  $(w_s \diamond p)(t, \tau) = 0$  for  $|s| \neq t - \tau + 1$ .

**Definition 2** (Impulse response representation, [29]). Let  $\mathfrak{Y}$  be an input-output map of the form (3). We say that  $\mathfrak{Y}$  has a impulse response representation (IRR) if there exist functions

$$\theta_{i,j,\mathfrak{Y}} : \mathcal{S}(\mathbb{I}_0^{n_p}) \mapsto \mathbb{R}^{n_y \times n_u}, \quad \eta_{i,\mathfrak{Y}} : \mathcal{S}(\mathbb{I}_0^{n_p}) \mapsto \mathbb{R}^{n_y}, \quad i, j \in \mathbb{I}_0^{n_p},$$

such that

1.  $\{\theta_{i,j,\mathfrak{Y}}\}_{i,j=0}^{n_p}, \{\eta_{i,\mathfrak{Y}}\}_{i=0}^{n_p}$  satisfy an exponential growth condition, i.e., there exist constants  $K, R > 0$  such that  $\forall s \in \mathcal{S}(\mathbb{I}_0^{n_p})$  with  $|s| \geq 1$ ,

$$\|\theta_{i,j,\mathfrak{Y}}(s)\|_F \leq K R^{|s|}, \quad \|\eta_{i,\mathfrak{Y}}(s)\|_F \leq K R^{|s|},$$

for all  $i, j \in \mathbb{I}_0^{n_p}$ , where  $\|\cdot\|_F$  denotes the Frobenius norm;

2. for each  $(u, p) \in \mathcal{U} \times \mathcal{P}$ ,  $t \in \mathbb{T}$ ,

$$\mathfrak{Y}(u, p)(t) = \begin{cases} (g_{\mathfrak{Y}} \diamond p)(t) + \int_0^t (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) d\delta, & \text{in CT} \\ (g_{\mathfrak{Y}} \diamond p)(t) + \sum_{\delta=0}^{t-1} (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) & \text{in DT} \end{cases}$$

where  $g_{\mathfrak{Y}} \diamond p : \mathbb{T} \rightarrow \mathbb{R}^{n_y}$  and  $h_{\mathfrak{Y}} \diamond p : \{(\tau, t) \in \mathbb{T} \times \mathbb{T} \mid \tau \leq t\} \rightarrow \mathbb{R}^{n_y \times n_u}$  are defined as follows. In the CT case,

$$(g_{\mathfrak{Y}} \diamond p)(t) = \sum_{\substack{i \in \mathbb{I}_0^{n_p} \\ s \in \mathcal{S}(\mathbb{I}_0^{n_p})}} \eta_{i,\mathfrak{Y}}(s) p_i(t) (w_s \diamond p)(t, 0),$$

$$(h_{\mathfrak{Y}} \diamond p)(\delta, t) = \sum_{\substack{i,j \in \mathbb{I}_0^{n_p} \\ s \in \mathcal{S}(\mathbb{I}_0^{n_p})}} \theta_{i,j,\mathfrak{Y}}(s) p_i(t) p_j(\delta) (w_s \diamond p)(t, \delta),$$

and, in the DT case,

$$(g_{\mathfrak{Y}} \diamond p)(t) = \sum_{\substack{i \in \mathbb{I}_0^{n_p} \\ s \in \mathcal{S}(\mathbb{I}_0^{n_p})}} \eta_{i,\mathfrak{Y}}(s) p_i(t) (w_s \diamond p)(t - 1, 0),$$

$$(h_{\mathfrak{Y}} \diamond p)(\delta, t) = \sum_{\substack{i,j \in \mathbb{I}_0^{n_p} \\ s \in \mathcal{S}(\mathbb{I}_0^{n_p})}} \theta_{i,j,\mathfrak{Y}}(s) p_i(t) p_j(\delta) (w_s \diamond p)(t - 1, \delta + 1).$$

The functions  $\{\theta_{i,j,\mathfrak{Y}}, \eta_{i,\mathfrak{Y}}\}_{i,j=0}^{n_p}$  will be referred to as sub-Markov parameters of  $\mathfrak{Y}$ .

From [29, Lemma 2] it follows that if  $\mathfrak{Y}$  has an IRR, then the sub-Markov parameters are uniquely defined by  $\mathfrak{Y}$ . Conversely, from [29, Lemma 2] it follows that if two input-output maps both have an IRR and their sub-Markov parameters are equal, then these two functions are equal too.

To provide a good intuition on the notion of the IRR, recall from [29] the following result: if  $\mathfrak{Y}$  can be realized by an LPV-SSA representation  $\Sigma$  of the form (1) from the initial state  $x_o$ , then  $\mathfrak{Y}$  has an IRR representation, and for all  $\tau \leq t \in \mathbb{T}$ ,  $p \in \mathcal{P}$ ,

$$\begin{aligned} (g_{\mathfrak{Y}} \diamond p)(t) &= \begin{cases} C(p(t))\Phi_p(t-1,0)x_o & \text{in DT} \\ C(p(t))\Phi_p(t,0)x_o & \text{in CT} \end{cases} \\ (h_{\mathfrak{Y}} \diamond p)(\tau, t) &= \begin{cases} C(p(t))\Phi_p(t-1, \tau+1)B(p(\tau)) & \text{in DT} \\ C(p(t))\Phi_p(t, \tau)B(p(\tau)) & \text{in CT} \end{cases} \end{aligned} \quad (5)$$

Here  $\Phi_p(t, \tau)$  is the fundamental matrix of the time-varying linear system

$$\xi x(t) = A(p(t))x(t),$$

i.e.,

$$\Phi_p(\tau, \tau) = I_{n_x}$$

and for all  $\tau \leq t \in \mathbb{T}$ ,

$$\frac{d}{dt}\Phi_p(t, \tau) = A(p(t))\Phi_p(t, \tau)$$

in CT and

$$\Phi_p(t+1, \tau) = A(p(t))\Phi_p(t, \tau)$$

in DT. In fact, according to [29, Lemma 3] the following lemma holds.

**Lemma 1.** *An input-output map  $\mathfrak{Y}$  is realized by  $\Sigma$  of the form (1) from the initial state  $x_o$ , if and only if  $\mathfrak{Y}$  has an IRR which satisfies for all  $i, j \in \mathbb{I}_0^{n_p}$  that*

$$\forall s \in \mathcal{S}(\mathbb{I}_0^{n_p}) : \eta_{i,\mathfrak{Y}}(s) = C_i A_s x_o, \quad \theta_{i,j,\mathfrak{Y}}(s) = C_i A_s B_j \quad (6)$$

where for  $s = \epsilon$ ,  $A_s$  denotes the identity matrix, and for  $s = s_1 s_2 \cdots s_n$  and  $s_1, s_2, \dots, s_n \in \mathbb{I}_0^{n_p}$ ,  $n > 0$ ,  $A_s = A_{s_n} A_{s_{n-1}} \cdots A_{s_1}$ .

Existence of an IRR of  $\mathfrak{Y}$  implies that  $\mathfrak{Y}$  is affine in  $u$  and can be represented as a convergent infinite sum of iterated integrals in CT, while, in DT,  $\mathfrak{Y}$  is a polynomial in  $p$ . In [29], it was shown that the sub-Markov parameters determine the input-output map uniquely. Moreover, in [29], it was shown that the input-output maps of an LPV-SSA representation admit an IRR, and the corresponding sub-Markov parameters can be expressed as products of the system matrices of the LPV-SSA representation.

To illustrate Definition 2, we recall from [29] the following example.



**Example 1.** Assume that  $\mathbb{P} = \mathbb{R}$ ,  $n_u = n_y = 1$  and let  $\mathfrak{Y}$  be an input-output map of the form (3) and assume it has an IRR. Then in DT, using that  $p_0(t) = 1$  for all  $t \in \mathbb{T}$ ,

$$\begin{aligned} (h_{\mathfrak{Y}} \diamond p)(2, 5) &= \theta_{0,0,\mathfrak{Y}}(00) + p(4)\theta_{0,0,\mathfrak{Y}}(01) + p(3)\theta_{0,0,\mathfrak{Y}}(10) + \\ &+ p(3)p(4)\theta_{0,0,\mathfrak{Y}}(11) + \cdots + p(2)p(5)\theta_{1,1,\mathfrak{Y}}(00) + p(2)p(5)p(4)\theta_{1,1,\mathfrak{Y}}(01) + \\ &p(2)p(5)p(3)\theta_{1,1,\mathfrak{Y}}(10) + p(2)p(5)p(3)p(4)\theta_{1,1,\mathfrak{Y}}(11) \\ (g_{\mathfrak{Y}} \diamond p)(2) &= \eta_{0,\mathfrak{Y}}(00) + p(1)\eta_{0,\mathfrak{Y}}(01) + p(0)\eta_{0,\mathfrak{Y}}(10) + p(0)p(1)\eta_{0,\mathfrak{Y}}(11) + \\ &p(2)\eta_{1,\mathfrak{Y}}(00) + p(2)p(1)\eta_{1,\mathfrak{Y}}(01) + p(2)p(0)p(1)\eta_{1,\mathfrak{Y}}(11). \end{aligned}$$

For CT,

$$\begin{aligned} (h_{\mathfrak{Y}} \diamond p)(2, 5) &= \theta_{0,0,\mathfrak{Y}}(\epsilon) + 3\theta_{0,0,\mathfrak{Y}}(0) + \theta_{0,0,\mathfrak{Y}}(1) \int_2^5 p(s)ds + \\ &+ \cdots + \theta_{0,0,\mathfrak{Y}}(101) \int_2^5 p(s_1) \int_2^{s_1} \int_2^{s_2} p(s_3)ds_3ds_2ds_1 + \cdots \\ &+ p(2)[\theta_{0,1,\mathfrak{Y}}(\epsilon) + 3\theta_{0,1,\mathfrak{Y}}(0) + \theta_{0,1,\mathfrak{Y}}(1)] \int_2^5 p(s)ds \\ &+ p(2)\theta_{0,1,\mathfrak{Y}}(101) \int_2^5 p(s_1) \int_2^{s_1} \int_2^{s_2} p(s_3)ds_3ds_2ds_1 + \cdots \\ (g_{\mathfrak{Y}} \diamond p)(2) &= \eta_{0,\mathfrak{Y}}(\epsilon) + 2\eta_{0,\mathfrak{Y}}(0) + \eta_{0,\mathfrak{Y}}(1) \int_0^2 p(s)ds + \\ &+ \cdots + \eta_{0,\mathfrak{Y}}(101) \int_0^3 p(s_1) \int_0^{s_1} \int_0^{s_2} p(s_3)ds_3ds_2ds_1 + \cdots \\ &+ p(2)\eta_{1,\mathfrak{Y}}(\epsilon) + 2p(2)\eta_{1,\mathfrak{Y}}(0) + p(2)\eta_{1,\mathfrak{Y}}(1) \int_0^2 p(s)ds + \\ &\cdots + p(2)\eta_{1,\mathfrak{Y}}(101) \int_0^2 p(s_1) \int_0^{s_1} \int_0^{s_2} p(s_3)ds_3ds_2ds_1 + \cdots \end{aligned}$$

That is, in DT,  $(h_{\mathfrak{Y}} \diamond p)(2, 5)$  is a polynomial of  $p(2), p(3), p(4), p(5)$ , and the degree of  $p(2), p(3), p(4), p(5)$  in each monomial is at most one. Moreover, for each  $i, j \in \{0, 1\}$ ,  $\theta_{i,j,\mathfrak{Y}}(s)$  are the coefficients of this polynomial with  $s$  being a sequence of elements  $\{0, 1\}$  of length at most 2. In particular, only the components of the sub-Markov parameters of the form  $\theta_{\mathfrak{Y}}(s)$ , with  $s$  being of length 2, occur in  $(h_{\mathfrak{Y}} \diamond p)(2, 5)$ . In contrast, in CT,  $(h_{\mathfrak{Y}} \diamond p)(2, 5)$  is an infinite sum of iterated integrals of  $p$ , where all the components of the form  $\theta_{i,j,\mathfrak{Y}}(s)$ ,  $i, j = 0, 1$ , with  $s$  being a sequence of arbitrary length, occur in  $(h_{\mathfrak{Y}} \diamond p)(2, 5)$ . The picture for  $(g_{\mathfrak{Y}} \diamond p)(2)$  is analogous.

**Example 2.** Assume that  $\mathbb{P} = \mathbb{R}$ ,  $n_u = n_y = 1$  and let  $\mathfrak{Y}$  be an input-output map of the form (3) and assume it has an IRR with  $\theta_{i,j,\mathfrak{Y}}(\epsilon) = 0$ ,

$$\theta_{1,j,\mathfrak{Y}}(l) = 0, \theta_{i,0,\mathfrak{Y}}(l) = 0, \eta_{1,\mathfrak{Y}}(l) = 0$$

for all  $i, j, l \in \mathbb{I}_0^1 = \{0, 1\}$ , and

$$\theta_{0,1,\mathfrak{Y}}(0) = 0, \theta_{0,1,\mathfrak{Y}}(1) = 1, \eta_{0,\mathfrak{Y}}(\epsilon) = 1, \eta_{0,\mathfrak{Y}}(0) = 0, \eta_{0,\mathfrak{Y}}(1) = 1$$

and for all  $w \in \mathcal{S}(\mathbb{I}_0^{n_p})$ ,  $|w| \leq 2$ ,  $\theta_{i,j,\mathfrak{Y}}(w)$ ,  $\eta_{i,\mathfrak{Y}}(w)$  are defined recursively w.r.t. the length of  $w$  as follows:

- if  $w = v00$  or  $w = v10$  for some  $v \in \mathcal{S}(\mathbb{I}_0^{n_p})$ , i.e.,  $w$  terminates with the sequence 00 or 10, then  $\theta_{i,j,\mathfrak{Y}}(w) = \eta_{i,\mathfrak{Y}}(w) = 0$ ;
- if  $w = v11$  for some  $v \in \mathcal{S}(\mathbb{I}_0^{n_p})$ , i.e.,  $w$  terminates with the sequence 11, then  $\theta_{i,j,\mathfrak{Y}}(w) = \theta_{i,j,\mathfrak{Y}}(v)$ ,  $\eta_{i,\mathfrak{Y}}(w) = \eta_{i,\mathfrak{Y}}(v)$ ;
- if  $w = v01$ , for some  $v \in \mathcal{S}(\mathbb{I}_0^{n_p})$ , i.e.,  $w$  terminates with the sequence 01, then  $\theta_{i,j,\mathfrak{Y}}(w) = \theta_{i,j,\mathfrak{Y}}(v1) + \theta_{i,j,\mathfrak{Y}}(v)$  and  $\eta_{i,\mathfrak{Y}}(w) = \eta_{i,\mathfrak{Y}}(v1) + \eta_{i,\mathfrak{Y}}(v)$ .

Then it is easy to see that  $\theta_{i,j,\mathfrak{Y}}$ ,  $\eta_{i,\mathfrak{Y}}$  satisfy (6) with

$$A_0 = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}, A_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, B_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, B_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ C_0 = [1 \ 0], C_1 = [0 \ 0], x_0 = [1 \ 1]^\top$$

and as the result  $\Sigma = (\mathbb{P}, \{A_i, B_i, C_i\}_{i=0}^1)$  is a realization of  $\mathfrak{Y}$ , both in DT and CT.

### 3. Statement of the main results

In this section, we state the main results of the paper. For the sake of readability, the proofs of the results will be given later in Section 4.

In order to state the main result, we have to define the class of input-output representations which correspond to LPV-SSA representations. Informally, the latter class will be the class of equations which are linear in the repeated shifts (DT) or the high-order derivatives (CT) of the inputs and the outputs, and which are polynomial in the repeated shifts (DT) or the high-order derivatives (CT) of the scheduling signal. In order for such representations to be well posed in the CT case, we have to show that smooth inputs and scheduling signals result in smooth output responses at least for input-output maps which admit an IRR. This will be done in the next lemma.

**Lemma 2.** *In CT, if a given input-output map  $\mathfrak{Y}$  of the form (3) admits an IRR, then  $\mathfrak{Y}(u, p)$  is smooth for all smooth  $(p, u) \in \mathcal{P} \times \mathcal{U}$ .*

*Proof of Lemma 2.* See Subsection 4.4. □

In order to define the desired class of input-output representations formally, we will use the following terminology: a function  $f : \mathbb{R}^m \rightarrow \mathbb{R}^{k \times l}$  will be said to be *polynomial*, if its coordinate functions are polynomial, i.e. if  $f_{i,j} : \mathbb{R}^m \ni x \mapsto (f(x))_{i,j} \in \mathbb{R}$  is a polynomial function for all  $i \in \{1, \dots, k\}$ ,  $j \in \{1, \dots, l\}$ .

**Definition 3** (LPV-ARX representation). *An LPV input-output autoregressive (abbreviated by LPV-ARX) representation is a collection*

$$(\{\mathcal{A}_{i,j}\}_{i \in \mathbb{I}_1^{n_y}, j \in \mathbb{I}_0^n}, \{\mathcal{B}_{i,j}\}_{i \in \mathbb{I}_1^{n_y}, j \in \mathbb{I}_0^{n-1}}), \quad (7)$$

of polynomial functions of the form  $\mathcal{A}_{i,j} : \mathbb{R}^{n_p(n+1)} \rightarrow \mathbb{R}$ ,  $i \in \mathbb{I}_1^{n_y}$ ,  $j \in \mathbb{I}_0^n$  and  $\mathcal{B}_{i,j} : \mathbb{R}^{n_p(n+1)} \rightarrow \mathbb{R}^{1 \times n_u}$ ,  $i \in \mathbb{I}_1^{n_y}$ ,  $j \in \mathbb{I}_0^{n-1}$ , such that, for each  $i \in \mathbb{I}_1^{n_y}$ , not all of the parameters  $\{\mathcal{A}_{i,j}\}_{i \in \mathbb{I}_1^n}$  are zero, i.e.,  $\mathcal{A}_{i,j} \neq 0$  for some  $j \in \mathbb{I}_i$ .

**Definition 4** (LPV-ARX representation of an input-output map). *Consider an input-output map  $\mathfrak{Y}$  of the form (3) that admits an IRR. The LPV-ARX representation (7) is said to be an LPV-ARX representation of  $\mathfrak{Y}$ , if for any pair of signals  $u \in \mathcal{U}, p \in \mathcal{P}$ , such that in CT case  $u$  and  $p$  are smooth, the signal  $y = \mathfrak{Y}(u, p)$  satisfies*

$$\sum_{j=0}^n \mathcal{A}_{i,j}(p, \xi p, \dots, \xi^n p) \xi^j y_i = \sum_{j=0}^{n-1} \mathcal{B}_{i,j}(p, \xi p, \dots, \xi^n p) \xi^j u, \quad (8)$$

for all  $i \in \mathbb{I}_1^{n_y}$ , where  $y_i$  is the  $i$ th entry of  $y$ ,  $i \in \mathbb{I}_1^{n_y}$ .

**Example 3** (Continuous-time LPV-ARX representation). *Consider the following LPV-ARX representation:  $n_p = 2$ ,  $n_y = 1$ ,  $n_u = 1$ ,  $n = 2$ ,  $\mathcal{A}_{1,2}(p, \frac{d}{dt}p) = p$ ,  $\mathcal{A}_{1,1}(p, \frac{d}{dt}p) = -(p + (\frac{d}{dt}p))$ ,  $\mathcal{A}_{1,0}(p, \frac{d}{dt}p) = -(1+p)p^2$ ,  $\mathcal{B}_{1,0}(p, \frac{d}{dt}p) = p^3$ ,  $\mathcal{B}_{1,1}(p, \frac{d}{dt}p) = 0$ . Then*

$$(\{\mathcal{A}_{1,j}\}_{j=0}^2, \{\mathcal{B}_{1,j}\}_{j=0}^1) \quad (9)$$

is an LPV-ARX representation of  $\mathfrak{Y}$ , if for all  $y = \mathfrak{Y}(u, p)$ , where  $u$  and  $p$  are smooth, the following holds:

$$\ddot{y}(t)p(t) - \dot{y}(t)(p(t) + \dot{p}(t)) - (1 + p(t))p^2(t)y(t) = p^3(t)u(t),$$

where  $\ddot{y}(t) = \frac{d^2}{dt^2}y(t)$ ,  $\dot{y}(t) = \frac{d}{dt}y(t)$ ,  $\dot{p}(t) = \frac{d}{dt}p(t)$ .

**Example 4** (Discrete-time LPV-ARX representation). *Consider the following LPV-ARX representation:  $n_p = 2$ ,  $n_y = 1$ ,  $n_u = 1$ ,  $n = 2$ ,  $\mathcal{A}_{1,2}(p, \xi p) = p$ ,  $\mathcal{A}_{1,1}(p, \xi p) = -\xi p$ ,  $\mathcal{A}_{1,0}(p, \xi p) = -(1+p)p\xi p$ ,  $\mathcal{B}_{1,0}(p, \xi p) = p^2\xi p$ ,  $\mathcal{B}_{1,1}(p, \xi p) = 0$ . Then*

$$(\{\mathcal{A}_{1,j}\}_{j=0}^2, \{\mathcal{B}_{1,j}\}_{j=0}^1) \quad (10)$$

is an LPV-ARX representation of  $\mathfrak{Y}$ , if for all  $y = \mathfrak{Y}(u, p)$ , the following holds:

$$y(t+2)p(t) - y(t+1)p(t+1) - (1+p(t))p(t)p(t+1)y(t) = p^2(t)p(t+1)u(t).$$

Note that Lemma 2 ensures that the right-hand side of (8) is well defined. We can now state the following theorem which is our main result.

**Theorem 1** (LPV-SSA representation of an input-output map). *Consider an input-output map  $\mathfrak{Y}$  which admits an IRR. Then  $\mathfrak{Y}$  has an LPV-SSA realization if and only if there exists an LPV-ARX representation of  $\mathfrak{Y}$ .*

*Proof of Theorem 1.* See Subsection 4.4. □

The main contribution of Theorem 1 is that it gives an exact characterization of those LPV-ARX representations which correspond to LPV-SSA realization. In [1], a more general form of LPV-ARX representations was considered. There, the coefficients of the input-output equations were assumed to be meromorphic functions of the future values of the scheduling variables (DT) or of the current derivatives of the scheduling variables (CT), and it was shown that an input-output map admits such a general LPV-ARX representation, if and only if it arises as an input-output map of a LPV state-space representation whose matrices depend on the scheduling variable in a meromorphic and dynamical manner. However, the results of [1] do not tell us which LPV-ARX representations correspond to LPV-SSA representations. In fact, even if the construction of [1] is applied to an LPV-SSA, and the resulting LPV-ARX representation is of the type defined in Definition 3, the application of the algorithm from [1, Section 4.3] will not in general yield an LPV-SSA, as it works with specific constructions of state basis, which often will not correspond to the state basis on which the original LPV-SSA representation was based on.

Theorem 1 implies that if we want to represent the input-output behavior of an LPV-SSA representation by an LPV-ARX, then in general, the coefficients of the LPV-ARX representation will still depend on scheduling variable in a dynamical way.

We prove Theorem 1 by using known results on the relationship between input-output equations and bilinear state-space representations. This relationship was explored in [24] in DT case, and in [25, 30, 31] for CT case. Note that bilinear state-space representations can be viewed as a subclass of LPV-SSA without control inputs, if we view scheduling signals as inputs, so the use of bilinear systems theory is quite natural in this context.

We have not formulated explicit algorithms for converting an LPV-ARX form to an LPV-SSA form or vice versa. The reason for this is that such algorithms would have to solve, as a special case, the problem of converting an input-output equation to a bilinear state-space representation and vice versa. However, despite the existence of a rich literature on bilinear systems, we have not found such algorithms in the literature. The results of [30, 31], which use differential algebra, suggest that in the CT case, input-output equations for bilinear state-space representations could, in principle, be obtained by computing formal derivatives of the inputs and outputs and then eliminating the state variable. However, no explicit algorithm has been formulated so far, to the best of our knowledge. The analysis of the proofs of [30, 31] does not suggest an obvious way to formulate such algorithms. The problem of designing algorithms for the opposite transformation, i.e., for converting input-output equations to bilinear state-space representations is not obvious either. The proofs of [24, 25, 30, 31] do not suggest any easy way to formulate such algorithms. For these reasons, the problem of formulating algorithms for converting an LPV-SSA to an LPV-ARX or vice versa remains a topic of future research.

This being said, the proof of Theorem 1 is partially constructive. In particu-

lar, it is possible to use the proof of Theorem 1 and the proofs from [24, 25, 30, 31] to perform the transformation from an LPV-ARX to an LPV-SSA and back for simple examples. Such an illustrating example is presented in Subsection 4.5 of Section 4.

#### 4. Proof of the main result

Below we present the proof of the main result of the paper formulated as Theorem 1. We prove Theorem 1 by first showing that it can be reduced to the single output case, and then using known results on the relationship between input-output equations and bilinear state-space representations. The section is organized as follows. In Subsection 4.1 we sketch the structure of the proof of Theorem 1. In Subsection 4.2 we present the argument that it is enough to prove Theorem 1 for the single output case. We recall the necessary facts for bilinear state-space representations in Subsection 4.3. Then in Subsection 4.4 we present the proof of Theorem 1 and Lemma 2. We conclude the section by presenting in Subsection 4.5 an example illustrating the steps of the proof of Theorem 1.

##### 4.1. Structure of the proof

The main ingredients of the proof are Lemma 5, Lemma 6 and Lemma 9 which will be stated in Subsection 4.4. In this section we aim at informally explaining the role of these lemmas in the proof of Theorem 1. Lemma 5-6 relate LPV-SSA realizations of  $\mathfrak{Y}$  with bilinear state-space representations of the input-output maps  $p \mapsto (g_{\mathfrak{Y}} \diamond p)$  and  $p \mapsto (h_{\mathfrak{Y}} \diamond p)$ . Lemma 9 relates LPV-ARX representations of  $\mathfrak{Y}$  with input-output equations for  $p \mapsto (g_{\mathfrak{Y}} \diamond p)$  and  $p \mapsto (h_{\mathfrak{Y}} \diamond p)$ . The latter input-output equations are of the same type as the ones which correspond to bilinear state-space representations by [24, 25, 30, 31].

More precisely, the proof consists of the following steps. Assume that  $\mathfrak{Y}$  is an input-output map admitting an IRR representation.

*Step 1: The relationship between LPV-SSA and bilinear state-space representations (Lemma 5 and Lemma 6).* First, in Lemma 5 we show that if  $\Sigma$  is a LPV-SSA of the form (1) which is a realization of  $\mathfrak{Y}$  from some initial state  $x_o$ , then we can construct a bilinear state-space representation  $\mathcal{B}$  from the matrices of  $\Sigma$  such that  $\mathcal{B}$  is driven by scheduling signals from  $\mathcal{P}$  as inputs, and the values of the functions  $(g_{\mathfrak{Y}} \diamond p)$  and  $(h_{\mathfrak{Y}} \diamond p)$  are the time derivatives (in CT) or time shifts (in DT) of the outputs of  $\mathcal{B}$  induced by the input  $p$  and suitably chosen initial states.

Conversely, in Lemma 6 we show that if there exists a bilinear state-space representation  $\mathcal{B}'$  which is driven by scheduling signals from  $\mathcal{P}$ , and for each  $p \in \mathcal{P}$ , the values of the functions  $(g_{\mathfrak{Y}} \diamond p)$  and  $(h_{\mathfrak{Y}} \diamond p)$  are the time derivatives (in CT) or time shifts (in DT) of the outputs of  $\mathcal{B}'$  induced by the input  $p$  and suitable initial states, then we can construct a LPV-SSA realizing  $\mathfrak{Y}$  from  $\mathcal{B}'$ .

*Step 2: LPV-ARX representations and input-output equations for  $(g_{\mathfrak{Y}} \diamond p)$  and  $(h_{\mathfrak{Y}} \diamond p)$  (Lemma 9)*. In Lemma 9 we relate the existence of an LPV-ARX representation of  $\mathfrak{Y}$  with the existence of an input-output equation for the maps  $p \mapsto (g_{\mathfrak{Y}} \diamond p)$  and  $p \mapsto (h_{\mathfrak{Y}} \diamond p)$ . More precisely, in Part I of Lemma 9 we show that if  $\mathfrak{Y}$  has an LPV-ARX representation of the form (7), then the maps  $p \mapsto (g_{\mathfrak{Y}} \diamond p)$  and  $p \mapsto (h_{\mathfrak{Y}} \diamond p)$  satisfy an input-output equation. The latter equation is linear in the derivatives (CT) or shifts (DT) of  $(g_{\mathfrak{Y}} \diamond p)$  and  $(h_{\mathfrak{Y}} \diamond p)$ , and it is polynomial in derivatives (CT) or shifts (DT) of  $p$ . Conversely, in Part II of Lemma 9 we show that if  $p \mapsto (g_{\mathfrak{Y}} \diamond p)$  and  $p \mapsto (h_{\mathfrak{Y}} \diamond p)$  both satisfy an input-output equation of the latter type, then we can construct an LPV-ARX representation of  $\mathfrak{Y}$  from the coefficients of that equation.

*Step 3: The proof of Theorem 1 for the direction ‘existence of LPV-ARX  $\implies$  existence of LPV-SSA’.* Assume that  $\mathfrak{Y}$  has an LPV-ARX representation of the form (7). The by Lemma 9 the functions  $(g_{\mathfrak{Y}} \diamond p)$  and  $(h_{\mathfrak{Y}} \diamond p)$  satisfy a suitable input-output equation. In fact, we can define input-output maps which act only on the scheduling signals from  $\mathcal{P}$ , such that

- the functions  $(g_{\mathfrak{Y}} \diamond p)$  and  $(h_{\mathfrak{Y}} \diamond p)$  are shifts (DT) or derivatives (CT) of output responses of these input-output maps to the input  $p$ ,
- these input-output maps satisfy input-output equations which are linear in the shifts (DT) or derivatives (CT) of the output, and polynomial in the shifts (DT) or derivatives (CT) of  $p$ .

From the second property using [24, 25, 30, 31] it follows that there exists a bilinear state-space representation which generates these input-output maps from suitable initial states. Then from Lemma 6 it follows that this bilinear state-space representation gives rise to an LPV-SSA realizing  $\mathfrak{Y}$ .

*Step 4: The proof of Theorem 1 for the direction ‘existence of LPV-SSA  $\implies$  existence of LPV-ARX’.* Assume that  $\Sigma$  is an LPV-SSA of the form (1) which is a realization of  $\mathfrak{Y}$  from some initial state  $x_o$ . Then from Lemma 5 it follows that there exists a bilinear state-space representation such that the functions  $(g_{\mathfrak{Y}} \diamond p)$  and  $(h_{\mathfrak{Y}} \diamond p)$  are the time derivatives (in CT) or time shifts (in DT) of the outputs of this bilinear state-space representations induced by the input  $p$  and suitable initial states. From [24, 25, 30, 31] it follows that input and output pairs of bilinear state-space representations satisfy input-output equation which are linear in the derivatives (CT) or shifts (DT) of the output and which are polynomial in the derivatives (CT) or shifts (DT) of the input. Then it can be shown that the maps  $p \mapsto (g_{\mathfrak{Y}} \diamond p)$  and  $p \mapsto (h_{\mathfrak{Y}} \diamond p)$  also satisfy an input-output equation of the same type. Then from Lemma 9 it follows that  $\mathfrak{Y}$  has an LPV-ARX representation.

#### 4.2. Reduction to the single output case

Note that it is enough to prove Theorem 1 for the single output case, i.e., when  $n_y = 1$ , as the general case follows from the single output one. Indeed,

consider any input-output map  $\mathfrak{Y}$  of the form (3), and let  $\mathfrak{Y}_i = e_i^\top \mathfrak{Y}$ ,  $i = 1, \dots, n_y$  be the elements of  $\mathfrak{Y}$ , i.e.,  $\forall u \in \mathcal{U}, p \in \mathcal{P}, t \in \mathbb{T}$ ,

$$\mathfrak{Y}(u, p)(t) = [\mathfrak{Y}_1(u, p)(t) \quad \dots \quad \mathfrak{Y}_{n_y}(u, p)(t)]^\top.$$

It is easy to see that  $\mathfrak{Y}$  has a realization by an LPV-SSA if and only if for each  $i = 1, \dots, n_y$ ,  $\mathfrak{Y}_i$  has a realization by an LPV-SSA. Furthermore, if

$$(\{\mathcal{A}_{i,j}\}_{i \in \mathbb{I}_1^{n_y}, j \in \mathbb{I}_0^n}, \{\mathcal{B}_{i,j}\}_{i \in \mathbb{I}_1^{n_y}, j \in \mathbb{I}_0^{n-1}})$$

is an LPV-ARX representation of  $\mathfrak{Y}$ , then for each  $i = 1, \dots, n_y$ , the LPV-ARX representation

$$(\{\mathcal{A}_{i,j}\}_{j \in \mathbb{I}_0^n}, \{\mathcal{B}_{i,j}\}_{j \in \mathbb{I}_0^{n-1}})$$

is an LPV-ARX representation of  $\mathfrak{Y}_i$ ,  $i \in \mathbb{I}_1^{n_y}$ . Conversely, assume that

$$(\{\mathcal{A}_j^i\}_{j \in \mathbb{I}_0^{n_i}}, \{\mathcal{B}_j^i\}_{j \in \mathbb{I}_0^{n_i-1}})$$

is an LPV-ARX representation of  $\mathfrak{Y}_i$  for all  $i \in \mathbb{I}_1^{n_y}$ . Then by taking  $n = \max\{n_i \mid i \in \mathbb{I}_1^{n_y}\}$  and defining

$$\mathcal{A}_{i,j} = \begin{cases} \mathcal{A}_j^i & i \leq n_i \\ 0 & \text{otherwise} \end{cases},$$

and

$$\mathcal{B}_{i,j} = \begin{cases} \mathcal{B}_j^i & i \leq n_i - 1 \\ 0 & \text{otherwise} \end{cases},$$

we obtain an LPV-ARX representation

$$(\{\mathcal{A}_{i,j}\}_{i \in \mathbb{I}_1^{n_y}, j \in \mathbb{I}_0^n}, \{\mathcal{B}_{i,j}\}_{i \in \mathbb{I}_1^{n_y}, j \in \mathbb{I}_0^{n-1}})$$

of  $\mathfrak{Y}$ . For this reason, *in the rest of the paper, we will assume that  $n_y = 1$ , i.e. we consider the single output case*

#### 4.3. Technical results for bilinear state-space representations

Recall from [24, 32, 33] that a bilinear state-space representation over the input-space  $\mathbb{P}$  is a system of the form

$$\begin{aligned} \xi z(t) &= N_0 z + \sum_{i=1}^{n_p} (N_i z(t)) p_i(t), \quad z(0) = z_o \\ y(t) &= C z(t) \end{aligned} \tag{11}$$

where  $N_i \in \mathbb{R}^{n_z \times n_z}$ ,  $i \in \mathbb{I}_0^{n_p}$ ,  $C \in \mathbb{R}^{1 \times n_z}$ . Let  $\mathcal{Z} = \mathcal{C}_a(\mathbb{R}_0^+, \mathbb{R}^{n_z})$ , in CT, and  $\mathcal{Z} = (\mathbb{R}^{n_z})^{\mathbb{N}}$  in DT. A solution of (11) is a tuple  $(z, y, p) \in \mathcal{Z} \times \mathcal{U} \times \mathcal{P}$  such that it satisfies (11). We will identify the bilinear system (11) with the tuple

$$(\{N_i\}_{i \in \mathbb{I}_0^{n_p}}, C, z_o).$$

The bilinear system (11) is said to be a *realization* of an input-output map  $F : \mathcal{P} \rightarrow \mathcal{Y}$ , if for every  $p \in \mathcal{P}$  there exists  $z \in \mathcal{Z}$ , such that  $(z, y, p)$  is a solution of (11) with  $y = F(p)$ . We recall from [32, 34] some technical facts on generating series (Fliess series) and their input-output maps. These facts will be used in the proof of the main result of the paper.

A *generating series over  $\mathbb{I}_0^{n_p}$*  is a function  $c : \mathcal{S}(\mathbb{I}_0^{n_p}) \rightarrow \mathbb{R}$  such that there exist  $K, R > 0$  which satisfies

$$\forall s \in \mathcal{S}(\mathbb{I}_0^{n_p}) : |c(s)| \leq KR^{|s|}.$$

The function  $F_c : \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R}^{n_p}) \rightarrow \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R})$  generated by a generating series  $c$  is defined as

$$F_c(p)(t) = \sum_{v \in \mathcal{S}(\mathbb{I}_0^{n_p})} c(v)(w_v \diamond p)(t, 0),$$

where  $w_s \diamond p$  is the sub-Markov dependence defined in Definition 1 for CT.

In the sequel, we follow the established tradition of [32, 34] and we will denote  $F_c(p)$  by  $F_c[p]$ . From [32], it follows that  $F_c$  is well defined. Note that the growth condition  $\forall s \in \mathcal{S}(\mathbb{I}_0^{n_p}) : c(s) \leq KR^{|s|}$  was necessary for  $F_c[u]$  to be well defined.

Next we extend the definition of generating series to include matrix and vector valued series. We define a *generating series* as a function  $c : \mathcal{S}(\mathbb{I}_0^{n_p}) \rightarrow \mathbb{R}^{n_r \times n_l}$  for some integers  $n_l, n_r > 0$ , such that there exist  $K, R > 0$ :

$$\forall v \in \mathcal{S}(\mathbb{I}_0^{n_p}) : \|c(v)\|_F \leq KR^{|v|}.$$

Here,  $\|\cdot\|_F$  denotes the Frobenius norm for matrices. If  $n_l = 1$ , then  $c$  is just a vector valued generating series. Let us denote by  $\mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R}^{n_r \times n_l})$  the set of all piecewise-continuous functions  $f : \mathbb{R}_0^+ \rightarrow \mathbb{R}^{n_r \times n_l}$ , i.e., functions such that for any  $i = 1, \dots, n_r, j = 1, \dots, n_l$ , the coordinate function  $f_{i,j} : s \mapsto (f(s))_{i,j}$  belongs to  $\mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R}^{n_r \times n_l})$ . We define  $F_c : \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R}^{n_p}) \rightarrow \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R}^{n_r \times n_l})$  as

$$F_c[p](t) = \sum_{v \in \mathcal{S}(\mathbb{I}_0^{n_p})} c(v)(w_v \diamond p)(t, 0),$$

where  $w_s \diamond p$  is the sub-Markov dependence defined in Definition 1 for CT, and the infinite summation is understood in the usual topology of matrices. Clearly, if  $c_{i,j}$  denotes the  $(i, j)$ th component of  $c$ ,  $c_{i,j}$  is a generating series in the classical sense and  $F_{c_{i,j}}[p](t)$  equals the  $(i, j)$ th entry of the matrix  $F_c[p](t)$ ,  $i = 1, \dots, n_r, j = 1, \dots, n_l$ . If  $n_l = 1$ , then  $F_c$  takes values in  $\mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R}^{n_r})$ , i.e.,  $F_c[p](t)$  is a vector in  $\mathbb{R}^{n_r}$ .

Although the map  $F_c$  was originally defined for CT, by an abuse of terminology, we can apply it in DT as well. If  $c : \mathcal{S}(\mathbb{I}_0^{n_p}) \rightarrow \mathbb{R}^{n_r \times n_l}$  is a generating series according to the definition above, then, in DT, we define the function generated by  $c$  as  $F_c : (\mathbb{R}^{n_p})^{\mathbb{N}} \rightarrow (\mathbb{R}^{n_r \times n_l})^{\mathbb{N}}$  such that

$$F_c(p)(t) = \sum_{v \in \mathcal{S}(\mathbb{I}_0^{n_p})} c(v)(w_v \diamond p)(t - 1, 0)$$



where  $w_s \diamond p$  is the sub-Markov dependence defined in Definition 1 for DT. Similarly to the CT case, we follow the established tradition of [32, 34] and we denote  $F_c(p)$  by  $F_c[p]$ . If  $n_l = 1$ , then  $F_c$  takes values in  $(\mathbb{R}^{n_r})^{\mathbb{N}}$ , i.e.,  $F_c[p](t)$  is a vector in  $\mathbb{R}^{n_r}$ .

Note that in contrast to the CT case, the definition of  $F_c[p](t)$  involves only a finite sum, and no growth condition on  $c$  is necessary in order to make  $F_c[p]$  well-defined. Note, however, that in this paper we will be interested in generating series  $c$  which potentially arise from bilinear state-space representations, and for such series the growth condition holds. The finiteness of the sum  $F_c[p]$  will allow us to use simpler proofs for the DT case.

It turns out that existence of a realization by a bilinear state-space representation is equivalent to existence of an input-output equation.

**Theorem 2** (Generating series based realisation, [24, 32, 33]). *A bilinear system of the form (11) is a realization of  $F : \mathcal{P} \rightarrow \mathcal{Y}$ , where  $\mathcal{Y} = \mathcal{C}_p(\mathbb{R}_0^+, \mathbb{R})$  in CT and  $\mathcal{Y} = (\mathbb{R})^{\mathbb{N}}$  in DT, if and only if there exists a convergent series  $c$  such that  $F = F_c$  and*

$$\forall s \in \mathcal{S}(\mathbb{I}_0^{n_p}) : c(s) = CN_s z_o,$$

where  $N_\epsilon = I_{n_z}$  and  $N_{q_1 \dots q_n} = N_{q_n} N_{q_{n-1}} \dots N_{q_1}$  for  $q_1, \dots, q_n \in \mathbb{I}_0^{n_p}$ ,  $n \geq 1$ .

This gives that input-output maps of bilinear systems are functions which arise from generating series. It turns out that input-output maps arising from generating series can be realized by a bilinear state-space form if and only if they satisfy an input-output equation of a certain form.

**Theorem 3** (Input-output equation form, [24, 25]). *Let  $c$  be a generating series. There exists a bilinear system of the form (11) which is a realization of  $F_c$  if and only if there exist an integer  $k \geq 1$  and polynomials  $E_0, E_1, \dots, E_k$  such that for any  $p \in \mathcal{P}$ ,  $p$  being smooth in CT case,*

$$\sum_{i=0}^k E_i(p, \xi p \dots, \xi^{k-1} p) \xi^i (F_c[p]) = 0. \quad (12)$$

Moreover, if  $F_c$  has a realization by a bilinear representation  $(\{N_i\}_{i=0}^{n_p}, C, z_o)$ , then the polynomials  $E_i$  which satisfy (12) can be chosen in such a way that they depend only on the matrices  $(\{N_i\}_{i=0}^{n_p}, C)$  and not on  $z_o$ .

Note that by [34, Eq. (6) and Lemmas 2,4], if  $p$  is smooth then so is  $F_c[p]$ . So for smooth  $p$ ,  $\xi^k F_c[p]$  in (12) is well posed. Note that in Theorem 3, the assumption that  $F$  arises from a generating series is essential. If this assumption is omitted, then there exist counter-examples of  $F$  which satisfy an input-output equation of the form (12), but which cannot be realized by a bilinear system, see [25].

For the proof of Theorem 1, we will need the following technical results.

**Notation 2** (Shift of a generating series). *If  $c : \mathcal{S}(\mathbb{I}_0^{n_p}) \rightarrow \mathbb{R}^{n_l \times n_r}$  is a generating series, then for every  $v \in \mathcal{S}(\mathbb{I}_0^{n_p})$ , we define the shift of  $c$  by  $v$  as  $v \circ c : \mathcal{S}(\mathbb{I}_0^{n_p}) \rightarrow \mathbb{R}^{n_l \times n_r}$  which corresponds to  $v \circ c(w) = c(vw)$ ,  $w \in \mathcal{S}(\mathbb{I}_0^{n_p})$ .*

**Lemma 3** (Derivative of functions generated by generating series). *Let  $c$  be a generating series. Then*

$$(\xi F_c[p])(t) = \sum_{q \in \mathbb{I}_0^{n_p}} p_q(t) F_{q \circ c}[p](t), \quad (13)$$

for all  $p \in \mathcal{P}$  in DT, and for all smooth  $p \in \mathcal{P}$  in CT.

Recall that for CT,  $(\xi F_c(p))(t) = \frac{d}{dt} F_c(p)(t)$ , and for DT,  $(\xi F_c(p))(t) = F_c(p)(t+1)$ .

*Proof of Lemma 3.* For CT, the statement of the lemma follows from [26, Eq. (13)]. For DT, it can be seen by a simple calculation. Indeed,  $(\xi F_c[p])(t) = F_c[p](t+1)$  and by definition  $F_c[p](t+1) = \sum_{s \in \mathcal{S}(\mathbb{I}_0^{n_p}), |s|=t+1} c(s)(w_s \diamond p)(t, 0)$ , since in DT,  $(w_s \diamond p)(t, 0) = 0$  of  $|s| \neq t+1$  for all  $s \in \mathcal{S}(\mathbb{I}_0^{n_p})$ . Since  $(w_s \diamond p)(t, 0) = p_q(t)(w_{s'} \diamond p)(t-1, 0)$  if  $s = s'q$ ,  $q \in \mathbb{I}_0^{n_p}$ ,  $s, s' \in \mathcal{S}(\mathbb{I}_0^{n_p})$ ,  $|s| = t+1$ ,  $|s'| = t$ , it follows that

$$\begin{aligned} F_c[p](t+1) &= \sum_{s \in \mathcal{S}(\mathbb{I}_0^{n_p}), |s|=t+1} c(s)(w_s \diamond p)(t, 0) = \\ &= \sum_{q \in \mathbb{I}_0^{n_p}} p_q(t) \underbrace{\sum_{s' \in \mathcal{S}(\mathbb{I}_0^{n_p}), |s'|=t} \underbrace{c(s'q)}_{=F_{q \circ c}[p](t)} (w_{s'} \diamond p)(t-1, 0)}_{=F_{q \circ c}[p](t)}. \end{aligned}$$

□

By repeated application of (13), it follows that for any  $k$  and  $v \in \mathcal{S}(\mathbb{I}_0^{n_p})$ ,  $0 < |v| \leq k$ , there exists a polynomial  $\alpha_{k,v}$  in variables  $\mathcal{T}_k = \{\mathcal{T}_{i,j}\}_{i=1,j=1}^{k,n_p}$  such that for any generating series  $c$ ,

$$\xi^k F_c[p](t) = \sum_{v \in \mathcal{S}(\mathbb{I}_0^{n_p}), 0 < |v| \leq k} \alpha_{k,v}(p(t), \dots, \xi^{k-1} p(t)) F_{v \circ c}[p](t). \quad (14)$$

Note that the polynomials  $\alpha_{k,v}$  are the same for any generating series  $c$ . However they depend on whether DT or CT is chosen. For example, for  $n_p = 1$ ,  $\alpha_{3,11} = 0$  and  $\alpha_{2,11}(p(t), \xi p(t)) = p_1(t+1)p_1(t)$  in DT and  $\alpha_{3,11}(p(t), \xi p(t), \xi^2 p(t)) = 3p_1(t)p_1(t)$ ,  $\alpha_{2,11}(p(t), \xi p(t)) = (p_1(t))^2$  in CT.

#### 4.4. Proof of Lemma 2 and Theorem 1

We will use Theorem 2 and Theorem 3 from the previous section to prove the main result of the paper. To this end, we have to introduce a number of definitions and to prove some technical results.

**Definition 5** (Generating series  $c^{\mathfrak{Y}}$  and  $c_{r,i}^{\mathfrak{Y}}, c_r^{\mathfrak{Y}}$ ). *Assume that  $\mathfrak{Y}$  has an IRR and define the convergent series  $c_r^{\mathfrak{Y}}$  with  $r \in \mathbb{I}_0^{n_p}$  as follows:*

$$c^{\mathfrak{Y}}(\epsilon) = 0, \quad c_r^{\mathfrak{Y}}(\epsilon) = 0,$$

and

$$c^{\mathfrak{Y}}(vq) = \eta_{q,\mathfrak{Y}}(v), \quad c_r^{\mathfrak{Y}}(vq) = \theta_{q,r,\mathfrak{Y}}(v).$$

Finally let  $c_{r,i}^{\mathfrak{Y}}$  be the generating series formed by the  $i$ th column of  $c_r^{\mathfrak{Y}}$ , for  $i = 1, \dots, n_u$ .

It turns out that every LPV-SSA yields bilinear systems of the form (11) realizing  $F_{c^{\mathfrak{Y}}}$ ,  $F_{c_{r,i}^{\mathfrak{Y}}}$  and vice versa.

**Lemma 4.** *There exists an LPV-SSA representation which realizes  $\mathfrak{Y}$ , if and only if the input-output maps  $F_{c^{\mathfrak{Y}}}$  and  $\{F_{c_{r,j}^{\mathfrak{Y}}}\}_{r \in \mathbb{I}_0^{n_p}, j \in \mathbb{I}_1^{n_u}}$  can be realized by bilinear systems of the form (11).*

Lemma 4 follows from the following two lemmas.

**Lemma 5.** *Assume that the LPV-SSA of the form (1) is a realization of  $\mathfrak{Y}$ . Define  $z_{o,0} = (x_o^\top, 0^\top)^\top$  and for all  $r \in \mathbb{I}_0^{n_p}$  and  $j \in \mathbb{I}_1^{n_u}$ , let  $z_{r,j} = (B_{r,i}^\top, 0^\top)^\top$ , where  $B_{r,i}$  is the  $i$ th column of  $B_r$  and let*

$$N_i = \begin{bmatrix} A_i & \mathbf{O}_{n_x,1} \\ C_i & 0 \end{bmatrix}, \quad i \in \mathbb{I}_0^{n_p}, \quad C = [0 \quad 0 \quad \dots \quad 0 \quad 1],$$

where  $\mathbf{O}_{n_x,1}$  is the  $n_x \times 1$  matrix all entries of which are zero. Then the bilinear system  $(\{N_i\}_{i=0}^{n_p}, C, z_{o,0})$  is a realization of  $F_{c^{\mathfrak{Y}}}$ , and the bilinear system  $(N_i, C, z_{r,i})$  is a realization of  $F_{c_{r,i}^{\mathfrak{Y}}}$  for all  $r \in \mathbb{I}_0^{n_p}$  and  $j \in \mathbb{I}_1^{n_u}$ .

*Proof of Lemma 5.* Notice that

$$Cz_o = Cz_{r,j} = 0, \quad CN_{sq} = [C_q A_s \quad 0],$$

for all  $s \in \mathcal{S}(\mathbb{I}_0^{n_p})^*$ ,  $q \in \mathbb{I}_0^{n_p}$ .

$$c^{\mathfrak{Y}}(\epsilon) = Cz_o = 0, \quad c^{\mathfrak{Y}r,j}(\epsilon) = Cz_{r,j} = 0.$$

Moreover, for any  $s \in \mathcal{S}(\mathbb{I}_0^{n_p})^*$ , using (6),

$$c^{\mathfrak{Y}}(sq) = \eta_{q,\mathfrak{Y}}(s) = C_q A_s x_o = CN_{sq} z_o,$$

and

$$c_r^{\mathfrak{Y}}(sq) = \theta_{q,r,\mathfrak{Y}}(s) = C_q A_s B_r.$$

Since  $c_{r,j}^{\mathfrak{Y}}(sq)$  is the  $j$ th column of  $c_r^{\mathfrak{Y}}(sq)$ ,

$$c_{r,j}^{\mathfrak{Y}}(sq) = C_q A_s B_{r,j} = CN_{sq} z_{r,j}.$$

The statement of the lemma follows now from Theorem 2.  $\square$

**Lemma 6.** *Assume that the bilinear system  $(\{N_i\}_{i=0}^{n_p}, C, z_o)$  is a realization of  $\mathfrak{Y}_{c^{\mathfrak{Y}}}$  and assume that the bilinear system  $(\{F_i^{r,j}\}_{i=0}^{n_p}, H^{r,j}, z_o^{r,j})$  is a realization of*

$\mathfrak{Y}_{c_{r,j}^{\mathfrak{y}}}$ . Define the integer  $n_x \geq 1$  and the matrices  $A_i \in \mathbb{R}^{n_x \times n_x}$ ,  $B_i \in \mathbb{R}^{n_x \times n_u}$ ,  $C_i \in \mathbb{R}^{1 \times n_x}$ ,  $i \in \mathbb{I}_0^{n_p}$ , and vector  $x_o \in \mathbb{R}^{n_x}$  such that

$$\begin{aligned} A_i &= \text{diag}(N_i, F_i^0, \dots, F_i^{n_p}), \\ C_i &= [C \quad H_0 \quad \dots \quad H_{n_p}] A_i, \\ [B_0 \quad \dots \quad B_{n_p}] &= \begin{bmatrix} \mathbf{O}_{n_o, (n_p+1)n_u} \\ \text{diag}(z_{o,0}, \dots, z_{o,n_p}) \end{bmatrix}, \\ x_o &= [z_o^\top \quad 0 \quad \dots \quad 0]^\top, \quad z_{o,r} = \text{diag}(z_o^{r,1}, \dots, z_o^{r,n_u}), \\ F_i^r &= \text{diag}(F_i^{r,1}, \dots, F_i^{r,n_u}), \quad H_r = [H^{r,1} \quad \dots \quad H^{r,n_u}], \end{aligned}$$

where  $n_o$  is the number of rows and columns of  $N_0$ ,  $\mathbf{O}_{n_o \times (n_p+1)n_u}$  is the  $n_o \times (n_p + 1)n_u$  matrix whose entries are all zero, and for any (not necessarily square) matrixes  $L_1, \dots, L_k$ ,  $k > 0$ ,  $\text{diag}(L_1, \dots, L_k)$  is the block diagonal matrix whose diagonal blocks are  $L_1, \dots, L_k$  in that order. Then the LPV-SSA  $(\mathbb{P}, \{A_i, B_i, C_i, 0\}_{i=0}^{n_p})$  is a realization of  $\mathfrak{Y}$  from  $x_o$ .

*Proof of Lemma 6.* Notice that for all  $s \in \mathcal{S}(\mathbb{I}_0^{n_p})$ ,  $q \in \mathbb{I}_0^{n_p}$ ,  $C_q A_s x_o = C N_{sq} z_o$  and the  $j$ th column of  $C_q A_s B_r$  equals to  $H^{r,j} F_{sq}^{r,j} z_o^{r,j}$ . From Theorem 2 and the definition of  $c^{\mathfrak{y}}$ ,  $c_{r,j}^{\mathfrak{y}}$  it follows that  $C N_{sq} z_o = c^{\mathfrak{y}}(sq) = \eta_{q,\mathfrak{y}}(s)$  and it follows that  $H^{r,j} F_{sq}^{r,j} z_o^{r,j} = c_{r,j}^{\mathfrak{y}}(sq)$  and the latter equals to the  $j$ th column of  $\theta_{q,r,\mathfrak{y}}(s)$ . Hence,  $(\mathbb{P}, \{A_i, B_i, C_i\}_{i=0}^{n_p})$  satisfies (6) and by Lemma 1 it is a realization of  $\mathfrak{Y}$ .  $\square$

Next, we show that there is a correspondence between LPV-ARX representations of  $\mathfrak{Y}$  and input-output equations of the form (12) for  $F_{c^{\mathfrak{y}}}$  and  $\{F_{c_{r,j}^{\mathfrak{y}}}\}_{r=0,j=1}^{n_p,n_u}$ .

**Lemma 7.** *Let  $p \in \mathcal{P}$ , and in CT, assume in addition that  $p$  is smooth. Then,*

$$\begin{aligned} (g_{\mathfrak{y}} \diamond p)(t) &= (\xi v)(t), \quad v(t) = F_{c^{\mathfrak{y}}} [p](t), \\ (h_{\mathfrak{y}} \diamond p)(\tau, t) &= \sum_{r \in \mathbb{I}_0^{n_p}} p_r(\tau) \xi v_{\tau,r}(t), \\ v_{\tau,r}(t) &= F_{c_r^{\mathfrak{y}}} [\sigma_{\tau_s}(p)](t - \tau_s), \quad r \in \mathbb{I}_0^{n_p}, \end{aligned}$$

where  $\tau_s = \tau$  in CT and  $\tau_s = \tau + 1$  in DT, and  $\sigma_{\tau_s}(p)(h) = p(h + \tau_s)$  for all  $h \in \mathbb{T}$ , and in DT we assume that  $t > \tau$ .

This means that the functions  $(g_{\mathfrak{y}} \diamond p)(t)$ ,  $(h_{\mathfrak{y}} \diamond p)(\tau, t)$  can be expressed via derivatives (in CT) or time shifts (in DT) of  $F_{c^{\mathfrak{y}}}$  and  $\{F_{c_{r,j}^{\mathfrak{y}}}\}_{r=0,j=1}^{n_p,n_u}$ .

*Proof of Lemma 7.* Notice that  $q \circ c^{\mathfrak{y}}$  equals to  $\eta_{q,\mathfrak{y}}$  while  $q \circ c_{r,j}^{\mathfrak{y}}$  equals to the  $j$ th column of  $\theta_{q,r,\mathfrak{y}}$  for all  $q \in \mathbb{I}_0^{n_p}$ . From the definition of the functions

$(g_{\mathfrak{Y}} \diamond p)(t)$ ,  $(h_{\mathfrak{Y}} \diamond p)(\tau, t)$  it follows that for all  $p \in \mathcal{P}$ ,  $\tau, t \in \mathbb{T}$ ,  $\tau \leq t$ ,  $j = 1, \dots, n_u$ ,

$$(g_{\mathfrak{Y}} \diamond p)(t) = \sum_{q \in \mathbb{I}_0^{n_p}} p_q(t) F_{q \circ c_{\mathfrak{Y}}} [p](t),$$

$$(h_{\mathfrak{Y}} \diamond p)_j(\tau, t) = \begin{cases} \sum_{q,r \in \mathbb{I}_0^{n_p}} p_r(\tau) p_q(t) F_{q \circ c_{\mathfrak{Y}}} [\sigma_{\tau}(p)](t - \tau) & \text{in CT,} \\ \sum_{q,r \in \mathbb{I}_0^{n_p}} p_r(\tau) p_q(t) F_{q \circ c_{\mathfrak{Y}}} [\sigma_{\tau+1}(p)](t - \tau) & \text{in DT, if } t > \tau, \end{cases}$$

where  $(h_{\mathfrak{Y}} \diamond p)_j(\tau, t)$  is the  $j$ th column of  $(h_{\mathfrak{Y}} \diamond p)(t, \tau)$ . Combining this with (13), we get the statement of the lemma.  $\square$

Now we are ready to present the proof of Lemma 2.

*Proof of Lemma 2.* Since  $\mathfrak{Y}$  admits an IRR, it is enough to show that if  $p$  and  $u$  are smooth, then  $g_{\mathfrak{Y}} \diamond p$ ,  $(h_{\mathfrak{Y}} \diamond p)(\tau, \cdot) : [\tau, +\infty) \ni t \mapsto (h_{\mathfrak{Y}} \diamond p)(\tau, t)$  are smooth functions. From [34, Eq. (6) and Lemmas 2,4] it follows that  $F_{c_{\mathfrak{Y}}} [p](t)$  and  $F_{c_{\mathfrak{Y}}} [\sigma_{\tau}(p)](t - \tau)$  are smooth in  $t$ , for all  $i, j \in \mathbb{I}_0^{n_p}$ , if  $p$  is smooth. From Lemma 7 it then follows that  $g_{\mathfrak{Y}} \diamond p$ ,  $(h_{\mathfrak{Y}} \diamond p)(\tau, \cdot)$  are smooth, if  $p$  is smooth.  $\square$

Next, note that for any  $p \in \mathcal{P}$  and  $u \in \mathcal{U}$ ,  $y = \mathfrak{Y}(u, p)$ , such that  $p$  and  $u$  are smooth if CT is considered,

$$(\xi^k y)(t) = \xi^k (g_{\mathfrak{Y}} \diamond p)(t) + \nu_k(p, u, t) + \begin{cases} \int_0^t \frac{d^k}{dt^k} (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) d\delta & \text{in CT} \\ \sum_{\delta=0}^{t-1} (h_{\mathfrak{Y}} \diamond p)(\delta, t+k) u(\delta) & \text{in DT,} \end{cases}$$

$$\nu_k(p, u, t) = \begin{cases} \sum_{j=0}^{k-1} \frac{d^{k-j-1}}{dt^{k-j-1}} \left( \left( \frac{d^j}{dt^j} (h_{\mathfrak{Y}} \diamond p)(\delta, t) \right) \Big|_{\delta=t} u(t) \right) & \text{in CT} \\ \sum_{\delta=t}^{t+k-1} (h_{\mathfrak{Y}} \diamond p)(\delta, t+k) u(\delta) & \text{in DT.} \end{cases} \quad (15)$$

In CT, we used the equality  $\frac{d}{dt} \int_0^t h(\tau, t) d\tau = h(t, t) + \int_0^t \frac{d}{dt} h(\tau, t) dt$  for any continuously differentiable function  $h$ . Note that by Lemma 7, the entries of  $\nu_k(u, p, t)$  are sums of products of  $\xi^{l_1} u(t)$  with  $l_1 = 1, \dots, k$  and expressions of the form

$$\frac{d^{l_2}}{dt^{l_2}} \left( \sum_{q=0}^{n_p} p_q(t) \left( \frac{d^{l_1+1}}{dt^{l_1+1}} F_{q \circ c_{\mathfrak{Y}, j}} [\sigma_{\delta}(p)](t - \delta) \right) \Big|_{\delta=t} \right), \quad l, l_2 = 1, \dots, k,$$

and

$$\sum_{q=0}^{n_p} p_q(t) F_{q \circ c_{\mathfrak{Y}, j}} [\sigma_{\delta+1}(p)](t+l), \quad l = 1, \dots, k,$$

in CT and DT respectively, and by (14), both expressions are polynomials in  $p(t), \xi p(t), \dots, \xi^k p(t)$ , where  $\xi = \frac{d}{dt}$  in CT and  $\xi$  is the forward shift operator in DT. Hence, we can state the following simple result.

**Lemma 8.** For all  $i \in \mathbb{N}$ , there exist polynomials  $\{R_{i,l,j}\}_{l=0,j=1}^{i-1,n_u}$  in  $(i+1)$  variables such that for any  $p \in \mathcal{P}$ ,  $u \in \mathcal{U}$ ,  $p, u$  smooth in CT,

$$\sum_{l=0}^{i-1} \sum_{j=1}^{n_u} R_{i,l,j}(p(t), \dots, \xi^i p(t)) \xi^l u_j(t) = \nu_i(p, u, t).$$

Moreover, for each  $i \in \mathbb{N}$  and  $l \in \mathbb{I}_0^{i-1}$ , there exist polynomials  $\{\beta_{i,l,v,r}\}_{v \in \mathcal{S}(\mathbb{I}_0^{n_p}), |v| \leq i, r \in \mathbb{I}_0^{n_p}}$  which do not depend on  $\{c_{r,j}^{\mathfrak{Y}}\}_{r \in \mathbb{I}_0^{n_p}, j \in \mathbb{I}_1^{n_u}}$ , such that for all  $\bar{p} \in \mathbb{R}^{i \cdot n_p}$ ,

$$R_{i,l,j}(\bar{p}) = \sum_{v \in \mathcal{S}(\mathbb{I}_0^{n_p}), |v| \leq i, r \in \mathbb{I}_0^{n_p}} c_{r,j}^{\mathfrak{Y}}(v) \beta_{i,l,v,r}(\bar{p}). \quad (16)$$

The polynomials can easily be computed based on  $\{\beta_{i,l,v,r}\}_{v \in \mathcal{S}(\mathbb{I}_0^{n_p}), |v| \leq i, r \in \mathbb{I}_0^{n_p}}$ , even without knowing  $\mathfrak{Y}$ , by using (14) and the fact that  $F_{v \circ c_{r,j}^{\mathfrak{Y}}}[\sigma_t(p)](0) = c_{r,j}^{\mathfrak{Y}}(v)$ . This implies that we can replace  $\xi^k y$  by  $\xi^{k+1} F_{c_{r,j}^{\mathfrak{Y}}}[p]$  and  $\xi^{k+1} F_{c_{r,j}^{\mathfrak{Y}}}[p]$  in the input-output equations. More precisely, the following technical lemma holds.

**Lemma 9.** Assume that  $\mathfrak{Y}$  has an IRR. Denote by  $(h_{\mathfrak{Y}} \diamond p)(\tau, \cdot)$  the function  $\{t \in \mathbb{T} \mid \tau \leq t\} \ni t \mapsto (h_{\mathfrak{Y}} \diamond p)(\tau, t)$ .

(I) If  $\mathfrak{Y}$  has an LPV-ARX representation (7), then for all  $p \in \mathcal{P}$ , with  $p$  being smooth in the CT case, for all  $v \in \{(g_{\mathfrak{Y}} \diamond p)\} \cup \{(h_{\mathfrak{Y}} \diamond p)(\tau, \cdot) \mid \tau \in \mathbb{T}\}$ ,

$$\sum_{i=0}^n A_{1,i}(p, \xi p, \dots, \xi^k p) \xi^i v = 0. \quad (17)$$

(II) Conversely, assume that there exist polynomials  $\{A_{1,i}\}_{i=0}^n$ , such that for all  $p \in \mathcal{P}$ , with  $p$  being smooth in the CT case, and for all  $v \in \{(g_{\mathfrak{Y}} \diamond p)\} \cup \{(h_{\mathfrak{Y}} \diamond p)(\tau, \cdot) \mid \tau \in \mathbb{T}\}$ , (17) holds. Define the polynomials  $\{\mathcal{B}_{1,j}\}_{j=0}^{n-1}$  as

$$\begin{aligned} \mathcal{B}_{1,l} &= [R_{l,1}, \dots, R_{l,n_u}], \quad l \in \mathbb{I}_0^{n-1}, \\ R_{l,j}(p, \xi p, \dots, \xi^n p) &= \sum_{i=l+1}^n A_{1,i}(p, \xi p, \dots, \xi^n p) R_{i,l,j}(p, \xi p, \dots, \xi^i p), \end{aligned} \quad (18)$$

where the polynomials  $\{R_{i,l,j}\}_{l=0,j=1}^{i-1,n_u}$  are as in Lemma 8. Then

$$(\{A_{1,i}\}_{i=0}^n, \{\mathcal{B}_{1,j}\}_{j=0}^{n-1})$$

is an LPV-ARX representation of  $\mathfrak{Y}$ .

Note that Part (II) of Lemma 9 is a rather straightforward consequence of (15), while the proof of Part (I) is more involved.

*Proof of Lemma 9. Part (I)*

Assume that (7) is an LPV-ARX representation of  $\mathfrak{Y}$ . If we take  $u = 0$ , then it follows that  $\xi^i \mathfrak{Y}(u, p)(t) = \xi^i(g_{\mathfrak{Y}} \diamond p)(t)$ , and hence from (8) it follows that

$$\sum_{i=0}^n A_{1,i}(p, \xi p, \dots, \xi^n p) \xi^i(g_{\mathfrak{Y}} \diamond p) = 0, \quad (19)$$

i.e. (17) holds for  $v = (g_{\mathfrak{Y}} \diamond p)$ . If we evaluate (8) for  $y = \mathfrak{Y}(u, p)$  at  $t \in \mathbb{T}$  such that  $\xi^i u(t) = 0$ ,  $i \in \mathbb{I}_0^{n-1}$  and we use (15) and (17) for  $v = g_{\mathfrak{Y}} \diamond p$  proven above, then we get

$$0 = \int_0^t \psi(\delta) u(\delta) d\delta \quad \text{in (CT)}, \quad 0 = \sum_{\delta=0}^{t-1} \psi(\delta) u(\delta) \quad \text{in (DT)}, \quad (20)$$

where

$$\psi(\delta) = \sum_{i=0}^n A_{1,i}(p(t), \xi p(t), \dots, \xi^n p(t)) \xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t)$$

and  $\xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t)$  denotes the result of applying the  $\xi^i$  operator to the function  $(h_{\mathfrak{Y}} \diamond p)(\delta, \cdot)$  and then evaluating the resulting function at  $t$ , i.e.,  $\xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t) = \frac{d^i}{dt^i}(h_{\mathfrak{Y}} \diamond p)(\delta, t)$  in CT, and  $\xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t) = (h_{\mathfrak{Y}} \diamond p)(\delta, t + i)$  in DT. We will show that (20) being true for any  $u$  such that  $\xi^i u(t) = 0$ ,  $i \in \mathbb{I}_1^n$  implies that  $\psi(\delta) = 0$  for all  $\delta \in [0, t]$ , i.e., that (17) holds for all  $v \in \{(h_{\mathfrak{Y}} \diamond p)(\tau, \cdot) \mid \tau \in \mathbb{T}\}$ .

It remains to show that  $\psi(\delta) = 0$ . For DT case, by choosing an  $u \in \mathcal{U}$  such that  $u(s) = e_j$ ,  $u(\delta) = 0$ ,  $\delta \neq s$  for  $s \in \mathbb{I}_0^{t-1}$ ,  $j \in \mathbb{I}_1^n$ , we get that (20) implies  $\psi(\delta) = 0$ .

For the CT case, [28, Chapter 11, Lemma 9.4] implies that that for any open interval  $I = [0, t'] \subseteq [0, t]$ ,  $t' < t$ , and any  $n \in \mathbb{N}$ , we can choose a smooth function  $\phi_n$  such that  $\phi_n$  is zero outside  $I$  (hence  $\frac{d^j}{dt^j} \phi_n(t) = 0$  for all  $j \in \mathbb{N}$ ),  $\phi_n$  converges to  $\chi_I$  in  $L_1([0, t])$ , where  $\chi_I$  denotes the indicator function of  $I$ . By taking  $u_n = \phi_n e_j$ , and using the first equation of (20),  $\int_I \psi(\delta) e_j d\delta = \lim_{n \rightarrow \infty} \int_0^t \psi(\delta) u_n(\delta) d\delta = 0$ , from which by [28, Chapter 11, Corollary 6.4] and continuity of  $\psi$  it follows that  $\psi(\delta) = 0$  for all  $\delta \in [0, t]$ .

**Part (II)** Assume that (17) is satisfied. Recall (15). From Lemma 8 and (15) it follows that that for  $y = \mathfrak{Y}(u, p)$ ,

$$\xi^i y = \eta_i + \begin{cases} \int_0^t \xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) d\delta, & \text{in CT,} \\ \sum_{\delta=0}^{t-1} \xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) & \text{in DT,} \end{cases}$$

$$\eta_i(t) = \xi^i(g_{\mathfrak{Y}} \diamond p)(t) + \sum_{l=0}^{i-1} \sum_{j=1}^{n_u} R_{i,l,j}(p(t), \dots, \xi^l p(t)) \xi^l u_j(t).$$

$\xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t)$  denotes the result of applying the  $\xi^i$  operator to the function  $(h_{\mathfrak{Y}} \diamond p)(\delta, \cdot)$  and then evaluating the resulting function at  $t$ , i.e.,  $\xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t) =$

$\frac{d^i}{dt^i}(h_{\mathfrak{Y}} \diamond p)(\delta, t)$  in CT, and  $\xi^i(h_{\mathfrak{Y}} \diamond p)(\delta, t) = (h_{\mathfrak{Y}} \diamond p)(\delta, t + i)$  in DT. Define the LPV-ARX representation  $(\{\mathcal{A}_i\}_{i=0}^n, \{\mathcal{B}_i\}_{i=0}^{n-1})$  as in (18). Then from the discussion above it follows that

$$\begin{aligned} \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \xi^i y(t) &= \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \xi^i (g_{\mathfrak{Y}} \diamond p)(t) + \\ &+ \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \sum_{l=0}^{i-1} \sum_{j=1}^{n_u} R_{i,l,j}(p(t), \dots, \xi^i p(t)) \xi^l u_j(t) + \\ &+ \begin{cases} \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \int_0^t \xi^i (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) d\delta, & \text{in CT,} \\ \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \sum_{\delta=0}^{t-1} \xi^i (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) & \text{in DT,} \end{cases} \end{aligned}$$

Notice that by (17),  $\sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \xi^i (g_{\mathfrak{Y}} \diamond p)(t) = 0$ , and in CT,

$$\begin{aligned} \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \int_0^t \xi^i (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) d\delta &= \\ \int_0^t \underbrace{\sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \xi^i (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta)}_{=0 \text{ by (17)}} d\delta &= 0 \end{aligned}$$

and in DT:

$$\begin{aligned} \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \sum_{\delta=0}^{t-1} \xi^i (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) &= \\ = \sum_{\delta=0}^{t-1} \underbrace{\sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \xi^i (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta)}_{=0 \text{ by (17)}} &= 0 \end{aligned}$$

Thus,

$$\begin{aligned} \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \xi^i y(t) &= \\ = \sum_{i=0}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) \sum_{l=0}^{i-1} \sum_{j=1}^{n_u} R_{i,l,j}(p(t), \dots, \xi^i p(t)) \xi^l u_j(t) &= \\ = \sum_{l=0}^{n-1} \sum_{j=1}^{n_u} \sum_{i=l+1}^n \mathcal{A}_{1,i}(p(t), \dots, \xi^n p(t)) R_{i,l,j}(p(t), \dots, \xi^i p(t)) \xi^l u_j(t) &= \\ = \sum_{l=0}^{n-1} \mathcal{B}_{1,l}(p(t), \dots, \xi^n p(t)) \xi^l u(t). \end{aligned}$$

That is, (8) holds, i.e.  $(\{\mathcal{A}_i\}_{i \in \mathbb{I}_1^{n_y}, j \in \mathbb{I}_0^n}, \{\mathcal{B}_i\}_{i \in \mathbb{I}_1^{n_y}, j \in \mathbb{I}_0^{n-1}})$  is an LPV-ARX representation of  $\mathfrak{Y}$ .  $\square$



**Lemma 10.** *With the notation of Lemma 9, if (17) holds for all  $v \in \{(g_{\mathfrak{y}} \diamond p)\} \cup \{(h_{\mathfrak{y}} \diamond p)(\tau, \cdot) \mid \tau \in \mathbb{T}\}$ , then and for all  $v \in \{F_{c_{\mathfrak{y}}} [p]\} \cup \{F_{c_r^{\mathfrak{y}}} [p] \mid r \in \mathbb{I}_0^{n_p}\}$ ,*

$$\sum_{i=1}^{n+1} A_{1,i-1}(p, \xi p, \dots, \xi^k p) \xi^i v = 0. \quad (21)$$

*Proof of Lemma 10.* Since by Lemma 7, if  $v = F_{c_{\mathfrak{y}}} [p]$ , then  $\xi v = (g_{\mathfrak{y}} \diamond p)(t)$ , and as  $(g_{\mathfrak{y}} \diamond p)$  satisfies (17), it follows that  $v = F_{c_{\mathfrak{y}}} [p]$  satisfies (21). If (17) holds, it follows that for all  $p \in \mathcal{P}$ ,  $\tau \geq 0$ ,  $\nu_{\tau,p} : \{t \in \mathbb{T} \mid t > \tau\} \ni t \mapsto (h_{\mathfrak{y}} \diamond p)(\tau, t)$  satisfies

$$\sum_{i=0}^n A_{1,i}(p, \xi p, \dots, \xi^k p) \xi^i \nu_{\tau,p} = 0, \quad (22)$$

for all  $\tau$  and  $p$ , such that  $p$  is smooth in case of CT. Assume (22) is true for all  $p \in \mathcal{P}$  (such that  $p$  is smooth in case of CT) and  $t \geq \tau$ . We argue that then (21) holds for all  $v \in \{F_{c_r^{\mathfrak{y}}} [p] \mid r \in \mathbb{I}_0^{n_p}\}$ .

First we consider the DT case. By Lemma 7,

$$\nu_{\tau,p}(t) = \sum_{r \in \mathbb{I}_0^{n_p}} p_r(\tau) \xi F_{c_r^{\mathfrak{y}}} [\delta_{\tau_s}(p)](t - \tau_s).$$

Since in DT,  $\tau_s = \tau + 1$ ,

$$\xi F_{c_r^{\mathfrak{y}}} [\delta_{\tau_s}(p)](t - \tau) = \xi F_{c_r^{\mathfrak{y}}} [\delta_{\tau+1}(p)](t - \tau)$$

does not depend on  $p(\tau)$ . Hence, replacing  $p$  by  $\hat{p}^r$  such that  $\hat{p}^r$  coincides with  $p$  on  $(0, t] \cap \mathbb{T}$  and  $\hat{p}^0(0) = 0$ ,  $\hat{p}^r(0) = e_r$ ,  $r = 1, \dots, n_p$ , it follows that

$$\nu_{0,\hat{p}^r}(t) = \xi F_{c_r^{\mathfrak{y}}} [\delta_1(\hat{p}^r)](t - 1) = \xi F_{c_r^{\mathfrak{y}}} [\delta_1(p)](t - 1).$$

Hence, for  $\nu(t) = F_{c_{\mathfrak{y}}} [\delta_1(p)](t)$ ,  $\nu_{0,\hat{p}^r}(t) = \xi \nu(t - 1)$ , and therefore  $\nu$  satisfies

$$\sum_{i=1}^{n+1} A_{1,i-1}(p(t), \xi p(t), \dots, \xi^k p(t)) \xi^i \nu(t - 1) = 0, \quad (23)$$

for all  $t \geq 1$ .

Let  $\tilde{p} = \delta_1(p)$ , then  $\xi^r \tilde{p}(t - 1) = \tilde{p}(t + r - 1) = p(t + r) = \xi^r p(t)$ . Then, it follows that (23) can be rewritten as

$$\sum_{i=1}^{n+1} A_{1,i-1}(\tilde{p}(t - 1), \xi \tilde{p}(t - 1), \dots, \xi^k \tilde{p}(t - 1)) \xi^i v(t - 1) = 0, \quad (24)$$

where  $v = F_{c_{\mathfrak{y}}} [\tilde{p}]$ . Note that any scheduling signal  $\tilde{p}$  arises as  $\delta_1(p) = \tilde{p}$  for a suitable  $p$ . Hence, as (24) holds for all scheduling signals  $\tilde{p}$  which are of the form  $\delta_1(p) = \tilde{p}$  for some  $p$ , it follows that (24) holds for all  $\tilde{p} \in \mathcal{P}$  and all  $t \geq 1$ . However, (24) is the same as (21), but with  $\tilde{p}$  instead of  $p$  and  $t - 1$  instead of

$t$ . Hence, if (24) holds for all  $\tilde{p} \in \mathcal{P}$  and  $t \geq 1$ , then (21) holds for all  $p \in \mathcal{P}$ , and  $t \geq 0$ , with  $\nu = \xi F_{c_r^y}[p]$ .

In CT, from [28, Chapter 11, Lemma 9.4] it follows that for any  $t > 0$  and any  $r = 0, \dots, n_p$ , we can choose  $0 < \epsilon < t$ , and a sequence of smooth functions  $\hat{p}^{n,r}$  such that

$$\lim_{n \rightarrow \infty} \int_0^t |p_j(s) - \hat{p}_j^{n,i}(s)| ds = 0, \quad j \in \mathbb{I}_1^{n_p}$$

and  $\hat{p}^{n,r}(0) = e_r$ , where  $e_0 = 0$ , and  $\hat{p}^{n,r}$  coincides with  $p$  on  $[\epsilon, +\infty)$ . Since

$$\nu_{0, \hat{p}^{n,r}} = \xi F_{c_r^y}[\hat{p}^{n,r}]$$

and, by assumption, (22) holds with  $p$  being replaced by  $\hat{p}^{n,r}$ , it follows that

$$\nu := \nu^{n,r} := F_{c_r^y}[\hat{p}^{n,e}](t)$$

satisfies (21) for all  $t > 0$  with  $p$  being replaced by  $\hat{p}^{n,r}$ . Notice that by [34, Lemma 2.2],  $F_{v \circ c_r^y}[p](t)$  is continuous in  $p$  in the  $L^1$  topology,<sup>1</sup> hence

$$\lim_{n \rightarrow \infty} F_{v \circ c_r^y}[\hat{p}^{n,r}](t) = F_{v \circ c_r^y}[p](t)$$

for all  $v \in \mathcal{S}(\mathbb{I}_0^{n_p})$ . Using this observation, (14) and  $\hat{p}^{n,r}(t) = p(t)$ , we get that

$$\lim_{n \rightarrow \infty} \xi^k F_{v \circ c_r^y}[\hat{p}^{n,r}](t) = \xi^k F_{v \circ c_r^y}[p](t)$$

for all  $k \geq 0$ . Moreover, clearly

$$A_{1,i}(\hat{p}^{n,r}(t), \dots, \xi^k \hat{p}^{n,r}(t)) = A_{1,i}((p(t), \xi p(t), \dots, \xi^k p(t)))$$

as  $\hat{p}^{n,r}$  coincides with  $p$  around  $t$ . By taking limits of (21) when applied to

$$\nu := \nu^{n,r} := F_{c_r^y}[\hat{p}^{n,r}](t)$$

and with  $p$  being replaced by  $\hat{p}^{n,r}$ , it follows that (21) holds for  $p$  and  $\nu = \xi^k F_{v \circ c_r^y}[p]$  for all  $t > 0$ . Finally, as  $p$  and  $F_{v \circ c_r^y}[p](t)$  are smooth in  $p$ , by letting  $t$  go to 0, it follows that (21) holds for all  $\nu = \xi^k F_{v \circ c_r^y}[p]$ , all smooth  $p$  and all  $t \geq 0$ .  $\square$

<sup>1</sup>More precisely, [34, Lemma 2.2] shows that for any  $t$  and any generating series  $c$ , and any real number  $T$ , the function  $F_c : \mathcal{V}_T \ni p \mapsto F_c[p]|_{[0,T]} \in \mathcal{C}[0,T]$  is continuous. Here  $\mathcal{C}[0,T]$  is the set of continuous functions defined on  $[0,T]$  with the topology induced by the form  $\|f\| = \sup_{t \in [0,T]} |f(t)|$ . The set  $\mathcal{V}_T$  is the set of functions  $p \in \mathcal{P}$  such that  $\|p\|_\infty = \sup_{t \in [0,T], i \in \mathbb{I}_1^{n_p}} |p_i(t)| < 1$  and it is considered with the norm  $L^1$  norm  $\|p\|_1 = \max_{i=1}^{n_p} \int_0^T |p_i(s)| ds$ . Note that [34, Lemma 2.2] can readily be extended to show that the function  $F_c : \mathcal{V}_T^R \ni p \mapsto F_c[p]|_{[0,T]} \in \mathcal{C}[0,T]$  is continuous. Here  $\mathcal{V}_T^R$  is the set of functions  $p \in \mathcal{P}$  such that  $\|p\|_\infty = \sup_{t \in [0,T], i \in \mathbb{I}_1^{n_p}} |p_i(t)| < R$ ,  $R > 0$ . As before,  $\mathcal{V}_T^R$  is considered with the  $L^1$  norm. To this end, it is sufficient to notice that  $p \in \mathcal{V}_T^R$  implies that  $p/R \in \mathcal{V}_T$ . Moreover, if we define the generating series  $\tilde{c}(s) = c(s)R^{l(s)}$ ,  $s \in \mathcal{S}(\mathbb{I}_0^{n_p})$ , where  $l(s)$  is the number of letters of  $s$  which are not 0, then  $F_{\tilde{c}}[p/R] = F_c[p]$ . Hence, continuity of  $F_c[p]$  in  $p$  follows from that of  $F_{\tilde{c}}[p/R]$  and the latter follows from [34, Lemma 2.2]. Finally, in order to apply this result in our setting, we can easily choose  $R$  such that  $p, \hat{p}^{n,i} \in \mathcal{V}_t^R$ .

Note that (17) is a special case of the input-output equation (12) described in Theorem 3. This then allows us to use Theorem 3 to relate realizability of  $\{F_{c^{\mathfrak{y}}}[p], F_{c_{r,i}^{\mathfrak{y}}} \mid r \in \mathbb{I}_0^{n_p}, i \in \mathbb{I}_1^{n_u}\}$  by a bilinear state-space representation with existence of an LPV-ARX representation of  $\mathfrak{Y}$ . Exploiting this relationship is the basic idea of the proof of Theorem 1 presented below.

*Proof of Theorem 1.* Consider the direction of implications separately.

**LPV-ARX  $\implies$  LPV-SSA:** Assume  $\mathfrak{Y}$  has an LPV-ARX representation. From Lemma 9 and Lemma 10, it then follows that  $c \in \{c^{\mathfrak{y}}, c_{r,i}^{\mathfrak{y}} \mid r \in \mathbb{I}_0^{n_p}, i \in \mathbb{I}_1^{n_u}\}$  satisfies (12) with  $E_0 = 0$  and  $E_k = A_{1,k-1}$ ,  $k \in \mathbb{I}_0^n$ . Hence, by Theorem 3,  $F_c$  has a realization by a bilinear system in state-space form. By Lemma 4, this implies that  $\mathfrak{Y}$  has a realization by an LPV-SSA.

**LPV-SSA  $\implies$  LPV-ARX:** Conversely, assume that  $\mathfrak{Y}$  has an LPV-SSA realization. Lemma 5 implies that there exist matrices  $\{N_i\}_{i=0}^{n_p}, C$  such that the bilinear system  $(\{N_i\}_{i=0}^{n_p}, C, z_{o,0})$  is a realization of  $F_{c^{\mathfrak{y}}}$ , and the bilinear system  $(\{N_i\}_{i=0}^{n_p}, C, z_{r,i})$  is a realization of  $F_{c_{r,i}^{\mathfrak{y}}}$  for all  $r \in \mathbb{I}_0^{n_p}$  and  $i \in \mathbb{I}_1^{n_u}$ , for suitable initial states  $z_{o,0}, z_{r,i}$ . From the second statement of Theorem 3, it follows that there exists polynomials  $E_0, \dots, E_k$  of  $kn_p$  variables, such that  $E_k \neq 0$  and

$$\sum_{i=0}^k E_i(p, \xi p, \dots, \xi^{k-1} p) \xi^i y = 0 \quad (25)$$

for all (in CT case, smooth)  $p \in \mathcal{P}$  and  $y \in \mathcal{Y}$  such that  $y = F_{c^{\mathfrak{y}}}[p]$  or  $y = F_{c_{r,j}^{\mathfrak{y}}}[p]$ ,  $j \in \mathbb{I}_1^{n_u}$ .

For the DT case, let us consider two cases. If  $E_0 \neq 0$ , then set  $n = k$ , and if  $E_0 = 0$  set  $n = k - 1$ . Let

$$\mathcal{A}_{1,i}(p, \xi p, \dots, \xi^n p) = \begin{cases} E_i(\xi p, \dots, \xi^n p) & \text{if } E_0 \neq 0 \\ E_{i+1}(p, \dots, \xi^n p) & \text{if } E_0 = 0 \end{cases} \quad (26)$$

$i = 0, \dots, n$ . If  $E_0 \neq 0$ , then let us apply  $\xi$  to both sides of (25). Then, it follows that  $\sum_{i=1}^{n+1} \mathcal{A}_{1,i-1}(p, \xi p, \dots, \xi^n p) \xi^i y = 0$  for any  $y = F_{c^{\mathfrak{y}}}[p]$  or  $y = F_{c_{r,j}^{\mathfrak{y}}}[p]$ ,  $j = 1, \dots, n_u$ . Hence, (17) indeed holds. If  $E_0 = 0$ , then (25) can be rewritten as  $\sum_{i=1}^{n+1} \mathcal{A}_{1,i-1}(p, \xi p, \dots, \xi^n p) \xi^i y = \sum_{i=1}^k E_i(p, \xi p, \dots, \xi^n p) \xi^i y = \sum_{i=0}^k E_i(p, \xi p, \dots, \xi^n p) \xi^i y = 0$  for any  $y = F_{c^{\mathfrak{y}}}[p]$  or  $y = F_{c_{r,j}^{\mathfrak{y}}}[p]$ ,  $j = 1, \dots, n_u$ . Hence, (17) again holds.

For the CT case, we again distinguish two cases:  $E_0 = 0$  and  $E_0 \neq 0$ . If  $E_0 = 0$ , then it follows that (17) holds with  $n = k - 1$ , and  $\mathcal{A}_{1,i} = E_{i+1}$ ,  $i = 0, \dots, n$ . If  $E_0 \neq 0$ , then (25) implies

$$E_0(p, \xi p, \dots, \xi^{k-1} p) y = - \sum_{i=1}^k E_i(p, \xi p, \dots, \xi^{k-1} p) \xi^i y. \quad (27)$$

By applying  $\xi$  to both sides of (25), and multiplying both sides by  $E_0(p, \xi p, \dots, \xi^{k-1} p)$ ,

and using (27) gives that

$$\sum_{i=0}^k \mathcal{A}_{1,i}(p, \xi p, \dots, \xi^k p) \xi^{i+1} y = 0$$

for unique polynomials  $\{\mathcal{A}_{1,i}\}_{i=0}^k$  such that for all  $p$ ,  $p$  smooth in CT,

$$\begin{aligned} \mathcal{A}_{1,k}(p, \dots, \xi^k p) &= E_0(p, \xi p, \dots, \xi^{k-1} p) E_{k-1}(p, \xi p, \dots, \xi^{k-1} p), \\ \mathcal{A}_{1,i}(p, \dots, \xi^k p) &= E_0(p, \xi p, \dots, \xi^{k-1} p) \times \\ &\times [-\xi E_0(p, \dots, \xi^k p) E_{i+1}(p, \xi p, \dots, \xi^{k-1} p) + \xi(E_{i+1}(p, \xi p, \dots, \xi^k p) + \\ &E_i(p, \dots, \xi^k p))], \end{aligned} \quad (28)$$

for all  $i = 0, \dots, k-1$ . Hence, (17) indeed holds.

According to Lemma 9, then there exists an LPV-ARX representation of  $\mathfrak{Y}$ .  $\square$

#### 4.5. Illustrating example

Assume that  $\mathfrak{Y}$  admits an IRR and that (9) (in CT) or (10) (in DT) is an LPV-ARX representation of  $\mathfrak{Y}$ . Furthermore, assume that  $\mathfrak{Y}(0, p)(0) = 1$  and  $\xi \mathfrak{Y}(0, p)(0) = p(0)$ . Then the proof of Theorem 1 implies that the LPV-SSA  $\Sigma = (\mathbb{P}, \{A_i, B_i, C_i\}_{i=0}^1)$ , where

$$\begin{aligned} A_0 &= \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}, A_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, B_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, B_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ C_0 &= [1 \ 0], C_1 = [0 \ 0] \end{aligned} \quad (29)$$

is realization of  $\mathfrak{Y}$  from the initial state  $x_o = [1 \ 1]^\top$ . Conversely, the proof of Theorem 1 implies that if the LPV-SSA  $\Sigma = (\mathbb{P}, \{A_i, B_i, C_i\}_{i=0}^1)$  which satisfies (29) is a realization of  $\mathfrak{Y}$  from  $x_o = [1 \ 1]^\top$ , then  $\mathfrak{Y}$  admits an LPV-ARX representation (9) (in CT) or (10) (in DT). Below we will elaborate on these claims.

*LPV-ARX of the form (9) (CT)  $\implies \Sigma$  is a realization of  $\mathfrak{Y}$ .* We will first demonstrate the conversion of an LPV-ARX to an LPV-SSA in the CT case. From Lemma 9 and Lemma 10 it follows that

$$\xi^3 y(t) p(t) - \xi^2 y(t) (p(t) + \xi p(t)) - (1 + p(t)) p^2(t) \xi y(t) = 0 \quad (30)$$

for all  $y \in \{F_{c^{\mathfrak{Y}}}[p]\} \cup \{F_{c_{i,1}^{\mathfrak{Y}}}[p] \mid i \in \mathbb{I}_1^{n_p}\}$ . For any  $c \in \{c^{\mathfrak{Y}}\} \cup \{c_{i,1}^{\mathfrak{Y}} \mid i \in \mathbb{I}_1^{n_p}\}$ , if  $y = F_c[p]$ , then there exist families of generating series

$$\begin{aligned} c_1(\mu) &= (0 \circ c) + \mu_1(1 \circ c) \\ c_2(\mu_1, \mu_2) &= (00 \circ c) + \mu_1(10 \circ c) + \mu_2(1 \circ c) + \mu_1(01 \circ c) + \mu_1^2(11 \circ c) \\ c_3(\mu_1, \mu_2, \mu_3) &= (000 \circ c) + \mu_1(100 \circ c) + \mu_2(10 \circ c) + (010 \circ c) \mu_1 + (110 \circ c) \mu_1^2 + \\ &\mu_3(1 \circ c) + \mu_2(01 \circ c) + \mu_2 \mu_1(11 \circ c) + \mu_2(01 \circ c) + \mu_1(001 \circ c) + \\ &+ \mu_1^2(101 \circ c) + 2\mu_1 \mu_2(11 \circ c) + \mu_1^2(011 \circ c) + \mu_1^3(111 \circ c) \end{aligned}$$

where  $\mu_1, \mu_2, \mu_3 \in \mathbb{R}$ , such that

$$\begin{aligned}\xi y(t) &= F_{c_1(p(t))}[p](t), \quad \xi^2 y(t) = F_{c_2(p(t), \xi p(t))}[p](t), \\ \xi^3 y(t) &= F_{c_3(p(t), \xi p(t), \xi^2 p(t))}[p](t).\end{aligned}\tag{31}$$

By substituting these expressions into (30) and using [34, Lemma 4.1], it follows that (30) is equivalent to

$$c_3(\mu_1, \mu_2, \mu_3)\mu_1 - c_1(\mu_1, \mu_2)(\mu_1 + \mu_2) - (1 + \mu_1)\mu_1^2 c_1(\mu_1) = 0, \tag{32}$$

for all  $\mu_1, \mu_2, \mu_3 \in \mathbb{R}$ . The left-hand side of (32) is a polynomial in  $\mu_1, \mu_2, \mu_3$ , and its coefficients are generating series. Since this polynomial is zero for any choice of  $\mu_1, \mu_2, \mu_3$ , it follows that the coefficients at each monomial in  $\mu_1, \mu_2, \mu_3$  in the left-hand side of (32) are zero. Notice that the coefficient for the monomial  $\mu_3\mu_1$  in the left-hand side of (32) is  $(1 \circ c)$ . Hence,  $(1 \circ c) = 0$  holds. This then implies that  $(v1 \circ c) = 0$  for all  $v \in \mathcal{S}(\mathbb{I}_0^{n_p})$ . Using this fact and the expressions for  $c_1(\mu_1), c_2(\mu_1, \mu_2), c_3(\mu_1, \mu_2)$ , by computing the coefficients for each monomial in  $\mu_1, \mu_2, \mu_3$  and setting them to zero, we derive the following equations:

$$0 \circ c = 110 \circ c, \quad 00 \circ c = 0, \quad 010 \circ c = (10 \circ c) + 0 \circ c, \quad 1 \circ c = 0 \tag{33}$$

Note that (33) holds for all  $c \in \{c^{\mathfrak{J}}\} \cup \{c_{i,1}^{\mathfrak{J}} \mid i \in \mathbb{I}_1^{n_p}\}$ . From (33) it follows that the linear span of  $\{v \circ c \mid v \in \mathcal{S}(\mathbb{I}_0^{n_p})\}$  is finite dimensional, and it is generated by  $\{0 \circ c, 10 \circ c, c\}$ . From [24, 32, 33] it then follows that for every  $c$  which satisfies (33),  $F_c$  can be realized by the bilinear system of the form  $(C, \{N_0, N_1\}, z_o(c))$ , where

$$N_0 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad N_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad z_o(c) = \begin{bmatrix} c(0) \\ c(10) \\ c(\epsilon) \end{bmatrix}, \quad C = [0 \quad 0 \quad 1] \tag{34}$$

To see that (34) is a realization of  $F_c$ , it is sufficient to verify that  $z(t) = (F_{0 \circ c}[p](t), F_{10 \circ c}[p](t), F_c[p](t))^\top$  satisfies the differential equation  $\dot{z}(t) = (N_0 + N_1 p(t))z(t)$ . The latter can be done using (13) and (33).

It is left to determine  $z_o(c)$  for all  $c \in \{c^{\mathfrak{J}}\} \cup \{c_{i,1}^{\mathfrak{J}} \mid i \in \mathbb{I}_1^{n_p}\}$ . To this end, notice that  $c(\epsilon) = 0$  by definition, hence it is left to find  $c(0)$  and  $c(10)$ .

Note that  $\mathfrak{Y}(0, p)(0) = c^{\mathfrak{J}}(0) + p(0)c^{\mathfrak{J}}(1) = 1$ . Since  $p$  is arbitrary, it implies that  $c^{\mathfrak{J}}(1) = 0$  and  $c^{\mathfrak{J}}(0) = 1$ . Using  $1 \circ c^{\mathfrak{J}} = 0$  and  $00 \circ c^{\mathfrak{J}} = 0$ , it is easy to see that  $\frac{d}{dt}\mathfrak{Y}(0, p)(t)|_{t=0} = c^{\mathfrak{J}}(10)p(0) = p(0)$  and the latter implies that  $c^{\mathfrak{J}}(10) = 1$ . Hence  $z_o(c^{\mathfrak{J}}) = [1 \quad 1 \quad 0]^\top$ .

By using the fact that the input-output map  $\mathfrak{Y}_0(u, p) = \mathfrak{Y}(u, p) - \mathfrak{Y}(0, p)$  also satisfies (9) and that  $\mathfrak{Y}_0(u, p)(t) = \int_0^t (h_{\mathfrak{Y}} \diamond p)(\delta, t)u(\delta)d\delta$ , and by evaluating

(9) at  $t = 0$ , it follows that

$$\begin{aligned}
p(0)v_2(0) - v_1(t)(p(0) + \dot{p}(0)) &= p^3(0)u(0), \\
v_1(0) &= (h_{\mathfrak{Y}} \diamond p)(0, 0)u(0), \\
v_2(0) &= \left( \frac{d}{dt}(h_{\mathfrak{Y}} \diamond p)(t, t) \right) \Big|_{t=0} u(0) + (h_{\mathfrak{Y}} \diamond p)(0, 0)\dot{u}(0) + \\
&+ \left( \frac{d}{dt}(h_{\mathfrak{Y}} \diamond p)(\tau, t) \right) \Big|_{\tau=t=0} u(0)
\end{aligned} \tag{35}$$

By applying (35) for the case when  $u(0) = 0$ ,  $\dot{u}(0) = 1$  it follows that for all smooth  $p \in \mathcal{P}$ ,  $(h_{\mathfrak{Y}} \diamond p)(0, 0) = 0$ . Using the fact that  $1 \circ c_{i,1}^{\mathfrak{Y}} = 0$  for  $i = 0, 1$ , it follows that  $(h_{\mathfrak{Y}} \diamond p)(0, 0) = c_{0,1}^{\mathfrak{Y}}(0) + c_{1,1}^{\mathfrak{Y}}(0)p(0) = 0$  for all  $p \in \mathcal{P}$ . The latter implies that  $c_{0,1}^{\mathfrak{Y}}(0) = 0$  and  $c_{1,1}^{\mathfrak{Y}}(0) = 0$ . Using these facts and  $1 \circ c_{i,1}^{\mathfrak{Y}} = 0$ ,  $i = 1, 2$  it follows that  $v_1(0) = 0$  and  $v_2(0) = (c_{0,1}^{\mathfrak{Y}}(10)p(0) + c_{1,1}^{\mathfrak{Y}}(10)p^2(0))u(0)$ . Substituting the latter expression into (34) leads to  $c_{0,1}^{\mathfrak{Y}}(10)p^2(0)u(0) + c_{1,1}^{\mathfrak{Y}}(10)p^3(0)u(0) = p^3(0)u(0)$ . As  $u(0)$  and  $p(0)$  can be chosen in an arbitrary manner, the latter equation implies  $c_{0,1}^{\mathfrak{Y}}(10) = 0$  and  $c_{1,1}^{\mathfrak{Y}}(10) = 1$ . Hence,  $z_o(c_{0,1}^{\mathfrak{Y}}) = 0$  and  $z_o(c_{1,1}^{\mathfrak{Y}}) = [0 \ 1 \ 0]^{\top}$ .

From  $(C, \{N_0, N_1\}, x_0(c))$ ,  $c \in \{c^{\mathfrak{Y}}, c_{i,1}^{\mathfrak{Y}}, i \in \mathbb{I}_1^{n_p}\}$  we can construct an LPV-SSA as described in Lemma 6 and then apply to it the minimization procedure from [29] to obtain  $\Sigma$ . Alternatively, we can apply Lemma 5 to  $\Sigma$  to get the bilinear systems  $(C, \{N_0, N_1\}, z_o(c))$ ,  $c \in \{c^{\mathfrak{Y}}, c_{i,1}^{\mathfrak{Y}}, i \in \mathbb{I}_1^{n_p}\}$

*LPV-ARX of the form (10) (DT)  $\implies \Sigma$  is a realization of  $\mathfrak{Y}$ .* Next, we illustrate the transition from LPV-ARX to a LPV-SSA in the DT case. The steps are similar to those of for the CT case. Indeed instead of (30), we can use the following equation

$$\xi^3 y(t)p(t) - \xi^2 y(t)\xi p(t) - (1 + p(t))p(t)\xi p(t)\xi y(t) = 0, \tag{36}$$

which holds for all  $y \in \{F_{c^{\mathfrak{Y}}}[p]\} \cup \{F_{c_{i,1}^{\mathfrak{Y}}}[p] \mid i \in \mathbb{I}_1^{n_p}\}$ , and (31) still holds for all  $c \in \{c^{\mathfrak{Y}}\} \cup \{c_{i,1}^{\mathfrak{Y}} \mid i \in \mathbb{I}_1^{n_p}\}$ , but

$$\begin{aligned}
c_1(\mu_1, \mu_2) &= (00 \circ c) + \mu_1(10 \circ c) + \mu_2(01 \circ c) + \mu_1\mu_2(11 \circ c) \\
c_3(\mu_1, \mu_2, \mu_3) &= (000 \circ c) + \mu_1(100 \circ c) + (010 \circ c)\mu_2 + (110 \circ c)\mu_1\mu_2 + \mu_3(001 \circ c) + \\
&+ \mu_1\mu_3(101 \circ c) + \mu_2\mu_3(011 \circ c) + \mu_1\mu_2\mu_3(111 \circ c)
\end{aligned}$$

In addition, instead of (32)

$$c_3(\mu_1, \mu_2, \mu_3)\mu_1 - c_1(\mu_1, \mu_2)\mu_2 - (1 + \mu_1)\mu_1\mu_2c_1(\mu_1) = 0, \quad \mu_1, \mu_2, \mu_3 \in \mathbb{R} \tag{37}$$

holds. Again, by regrouping the terms with the same monomial in  $\mu_1, \mu_2, \mu_3$  and equating these coefficients to zero, we can derive the same inequalities as in (33).

Similarly to the CT case, we can derive a bilinear system  $(C, \{N_0, N_1\}, z_o(c))$  satisfying (34) which is a realization of  $F_c$  for  $c \in \{c^{\mathfrak{Y}}\} \cup \{c_{i,1}^{\mathfrak{Y}} \mid i \in \mathbb{I}_1^{n_p}\}$ .

It is then left to compute  $z_o(c)$ , for which we have to compute  $c(0)$  and  $c(10)$ , for  $c \in \{c^{\mathfrak{Y}}\} \cup \{c_{i,1}^{\mathfrak{Y}} \mid i \in \mathbb{I}_1^{n_p}\}$ . Similarly to the CT case, we first compute  $z_o(c)$  for  $c = c^{\mathfrak{Y}}$ . To this end, using  $1 \circ c^{\mathfrak{Y}} = 0$  it follows that  $\mathfrak{Y}(0, p)(0) = c^{\mathfrak{Y}}(0)$  and  $\xi\mathfrak{Y}(0, p)(0) = c^{\mathfrak{Y}}(00) + p(0)c^{\mathfrak{Y}}(10)$ . Hence, from  $\mathfrak{Y}(0, p)(0) = 1$  and  $\xi\mathfrak{Y}(0, p)(0) = p(0)$ , it follows that  $c^{\mathfrak{Y}}(0) = 1$ ,  $c^{\mathfrak{Y}}(10) = 1$ , and therefore  $z_o(c^{\mathfrak{Y}}) = [1 \ 1 \ 0]^T$ .

Next, we determine the values  $c_{0,1}^{\mathfrak{Y}}(0)$ ,  $c_{1,1}^{\mathfrak{Y}}(0)$ ,  $c_{0,1}^{\mathfrak{Y}}(10)$  and  $c_{1,1}^{\mathfrak{Y}}(10)$ . To this end, we proceed as in the CT case: we notice that  $\mathfrak{Y}_0(u, p) = \mathfrak{Y}(u, p) - \mathfrak{Y}(0, p)$  also satisfies (10) and that  $\mathfrak{Y}_0(u, p)(t) = \sum_{\delta=0}^{t-1} (h_{\mathfrak{Y}} \diamond p)(\delta, t) u(\delta) d\delta$ . By evaluating (10) at  $t = 0$  we get

$$\begin{aligned} p(0)v_2(0) - v_1(0)p(1) &= p^2(0)p(1)u(0), \\ v_1(0) &= (h_{\mathfrak{Y}} \diamond p)(0, 1)u(0), \\ v_2(0) &= (h_{\mathfrak{Y}} \diamond p)(1, 2)u(1) + (h_{\mathfrak{Y}} \diamond p)(0, 2)u(0) \end{aligned} \quad (38)$$

It follows that  $(h_{\mathfrak{Y}} \diamond p)(1, 2) = 0 = c_{0,0}^{\mathfrak{Y}}(0) + c_{1,0}^{\mathfrak{Y}}(1)p(1)$  (we used  $1 \circ c_{i,1}^{\mathfrak{Y}} = 0$  for  $i = 0, 1$ ). As  $p$  is arbitrary,  $c_{0,0}^{\mathfrak{Y}}(0) = c_{1,0}^{\mathfrak{Y}}(0) = 0$ . Then  $v_1(0) = (h_{\mathfrak{Y}} \diamond p)(0, 1)u(0) = 0$  and  $v_2(0) = (h_{\mathfrak{Y}} \diamond p)(0, 2)u(0) = c_{0,1}^{\mathfrak{Y}}(00)u(0) + c_{0,1}^{\mathfrak{Y}}(10)p(1)u(0) + c_{1,0}^{\mathfrak{Y}}(00)p(0)u(0) + p(0)p(1)u(0)c_{1,0}^{\mathfrak{Y}}(10)$ . By substituting the expression for  $v_2(0)$  into (38) and using the fact that  $p(0), p(1), u(0)$  can be arbitrary values, it follows that  $c_{0,1}^{\mathfrak{Y}}(10) = 0$  and  $c_{1,1}^{\mathfrak{Y}}(10) = 1$ . Hence, similarly to the CT case,  $z_o(c_{0,1}^{\mathfrak{Y}}) = 0$  and  $z_o(c_{1,1}^{\mathfrak{Y}}) = [0 \ 1 \ 0]^T$ .

As in the CT case, we can apply Lemma 6 to these bilinear systems and apply the minimization procedure from [29] to obtain  $\Sigma$ . As in CT, if Lemma 5 is applied to  $\Sigma$ , we get the bilinear systems  $(C, \{N_0, N_1\}, z_o(c))$ ,  $c \in \{c^{\mathfrak{Y}}\} \cup \{c_{i,1}^{\mathfrak{Y}} \mid i \in \mathbb{I}_1^{n_p}\}$

*LPV-SS of the form (29)  $\implies$  LPV-ARX representation (9) (CT) or (10) (DT).* Conversely, the proof of Theorem 1 implies that if  $\Sigma$  is a realization of an input-output map  $\mathfrak{Y}$ , then, in CT, (9) is an LPV-ARX representation of  $\mathfrak{Y}$ , and, in DT, (10) is an LPV-ARX representation of  $\mathfrak{Y}$ . To see this, let us construct the bilinear state-space representation described in Lemma 6. It then follows that the matrices  $C, N_0, N_1$  of that state-space representation satisfy (34).

We will argue that if  $c$  is any generating series such that  $(C, \{N_0, N_1\}, z_o(c))$  is a realization of  $F_c$ , then  $c$  satisfies the equations in (33). Indeed, it is easy to see  $v \circ c(s) = CN_v N_s z_o(c)$  for any  $s, v \in \mathcal{S}(\mathbb{I}_0^{n_p})$ . Since  $CN_1 = 0, CN_0^2 = 0, CN_0 N_1^2 = CN_0, CN_0 N_1 N_0 = CN_0 N_1 + CN_0$ , it follows that (33) holds.

That is, (33) holds for all  $c \in \{c^{\mathfrak{Y}}\} \cup \{c_{i,1}^{\mathfrak{Y}} \mid i \in \mathbb{I}_1^{n_p}\}$ . But (33) implies that (32) in CT and (37) in DT, which are equivalent to (30) in CT and (36) in DT respectively, for all  $y \in \{F_{c^{\mathfrak{Y}}}[p]\} \cup \{F_{c_{i,1}^{\mathfrak{Y}}}[p], i \in \mathbb{I}_1^{n_p}\}$ .

That is, (25) from the proof of Theorem 1 holds with  $k = 3$ ,  $E_0 = 0$ , and  $E_1 = -(1 + p)p^2$ ,  $E_2 = -(p + \xi p)$ ,  $E_3 = p$  in CT case and  $E_1 = -(1 + p)\xi p$ ,  $E_2 = -\xi p$ ,  $E_3 = p$  in DT case. Then using the proof of Theorem 1 it follows that (17) holds with  $\mathcal{A}_{j,1}$ ,  $j = 0, 1, 2$  as in (9) in CT and (10) in DT.

From the proof of Part (II) of Lemma 9, more precisely, the construction of  $\mathcal{B}_{j,1}$ ,  $i = 0, 1$  it follows that LPV-ARX (9) in CT and (10) in DT are LPV-ARX representations of  $\mathfrak{M}$ .

## 5. Conclusions

In this paper, we have studied the relationship between input-output equations and LPV-SSA representations. More precisely, we have shown that an input-output map can be realized by an LPV-SSA, if and only if it has a so called LPV-ARX representation, i.e., it satisfies certain input-output equations. This relationship is expected to be useful for system identification, as in principle it allows us to replace identification of LPV-SSA by LPV-ARX representations. The latter is in general simpler to achieve. In order to pursue this path, we will need algorithms for transforming an LPV-SSA representation to an LPV-ARX representation, and vice versa. Finding such algorithms will be a topic of future research. As it was mentioned before, finding such algorithms even for the special case of bilinear systems remains an open topic.

## References

- [1] R. Tóth, Modeling and Identification of Linear Parameter-Varying Systems, Lecture Notes in Control and Information Sciences, Vol. 403, Springer, Heidelberg, 2010.
- [2] R. Tóth, J. C. Willems, P. S. C. Heuberger, P. M. J. Van den Hof, The behavioral approach to linear parameter-varying systems, IEEE Trans. on Automatic Control 56 (2011) 2499–2514.
- [3] W. Rugh, J. S. Shamma, Research on gain scheduling, Automatica 36 (10) (2000) 1401–1425.
- [4] A. Packard, Gain scheduling via linear fractional transformations, Systems & Control Letters 22 (2) (1994) 79–92.
- [5] P. Apkarian, P. Gahinet, A convex characterization of gain-scheduled  $\mathcal{H}_\infty$  controllers, IEEE Trans. on Automatic Control 40 (5) (1995) 853–864.
- [6] 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) (1996) 929–952.
- [7] B. Lu, F. Wu, Switching LPV control designs using multiple parameter-dependent Lyapunov functions, Automatica 40 (11) (2004) 1973–1980.



- [8] F. Wu, K. Dong, Gain-scheduling control of LFT systems using parameter-dependent Lyapunov functions, *Automatica* 42 (1) (2006) 39–50.
- [9] C. W. Scherer, Robust controller synthesis is convex for systems without control channel uncertainties, in: *Model-Based Control: Bridging Rigorous Theory and Advanced Technology*, Springer, 2009, pp. 13–31.
- [10] R. Tóth, P. S. C. Heuberger, P. M. J. Van den Hof, Prediction error identification of LPV systems: present and beyond, in: J. Mohammadpour, C. W. Scherer (Eds.), *Control of Linear Parameter Varying Systems with Applications*, Springer, Heidelberg, 2012, pp. 27–60.
- [11] V. Laurain, M. Gilson, R. Tóth, H. Garnier, Refined instrumental variable methods for identification of LPV Box-Jenkins models, *Automatica* 46 (6) (2010) 959–967.
- [12] B. Bamieh, L. Giarré, Identification of linear parameter varying models, *Int. Journal of Robust and Nonlinear Control* 12 (2002) 841–853.
- [13] M. Butcher, A. Karimi, R. Longchamp, On the consistency of certain identification methods for linear parameter varying systems, in: *Proc. of the 17th IFAC World Congress, Seoul, Korea, 2008*, pp. 4018–4023.
- [14] K. Hsu, T. L. Vincent, K. Poolla, Nonparametric methods for the identification of linear parameter varying systems, in: *Proc. of the Int. Symposium on Computer-Aided Control System Design, San Antonio, Texas, USA, 2008*, pp. 846–851.
- [15] D. Vizer, G. Mercère, O. Prot, E. Laroche, M. Lovera, Linear fractional LPV model identification from local experiments: an  $H_\infty$ -based optimization technique, in: *Proc. Conference on Decision and Control*, 2013.
- [16] D. Vizer, G. Mercère, E. Laroche, O. Prot, Linear fractional LPV model identification from local experiments using an  $H_\infty$ -based local approach, in: M. Lovera (Ed.), *Control-oriented modelling and identification: theory and practice*, The Institution of Engineering and Technology, 2014.
- [17] J. W. van Wingerden, M. Verhaegen, Subspace identification of bilinear and LPV systems for open- and closed-loop data, *Automatica* 45 (2) (2009) 372–381.
- [18] P. L. dos Santos, J. A. Ramos, J. L. M. de Carvalho, Identification of LPV systems using successive approximations, in: *Proc. of the 47th IEEE Conf. on Decision and Control, Cancun, Mexico, 2008*, pp. 4509–4515.
- [19] M. Sznaier, C. Mazzaro, An LMI approach to the identification and (in)validation of LPV systems, in: S. Moheimani (Ed.), *Perspectives in robust control*, Vol. 268 of *Lecture Notes in Control and Information Sciences*, Springer, London, 2001, pp. 327–346.

- [20] V. Verdult, M. Verhaegen, Subspace identification of multivariable linear parameter-varying systems, *Automatica* 38 (5) (2002) 805–814.
- [21] M. Petreczky, G. Mercère, Affine LPV systems: realization theory, input-output equations and relationship with linear switched systems, in: Proc. IEEE Conference on Decision and Control, 2012.
- [22] M. Petreczky, G. Mercère, R. Tóth, Affine LPV systems: realization theory, input-output equations and relationship with linear switched systems, Tech. rep., arXiv 1209.0345 (2012).
- [23] E. Sontag, Polynomial response maps, Vol. 13, Springer Verlag, 1979.
- [24] E. Sontag, Realization theory of discrete-time nonlinear systems: Part i - the bounded case, *IEEE Transactions on Circuits and Systems* 26 (1979) 342–356.
- [25] E. D. Sontag, Bilinear realizability is equivalent to existence of a singular affine differential i/o equation, *Systems Control Lett.* 11 (3) (1988) 181–187.
- [26] Y. Wang, E. Sontag, On two definitions of observation spaces, *Systems and Control Letters* 13 (1989) 279–289.
- [27] J. Belikov, U. Kotta, M. Tonso, Comparison of {LPV} and nonlinear system theory: A realization problem, *Systems & Control Letters* 64 (2014) 72 – 78.
- [28] S. Lang, *Real Analysis*, Addison-Wesley, 1983.
- [29] M. Petreczky, R. Tóth, G. Mercère, Realization theory for LPV state-space representations with affine dependence, *IEEE Transactions on Automatic Control* 62 (2017) 4667–4674.
- [30] M. Fliess, C. Reutenauer, Une application de l’algèbre différentielle aux systèmes réguliers (ou bilinéaires), in: A. Bensoussan, J. L. Lions (Eds.), *Analysis and Optimization of Systems*, Vol. 44 of Lect. Notes Control Inform. Sci., Springer-Verlag, Berlin, 1982, pp. 99 – 107.
- [31] M. Fliess, C. Reutenauer, Théorie de picard-vessiot des systèmes réguliers (ou bilinéaires), in: *Outils et Modèles Mathématiques pour L’Automatique L’Analysis de Systèmes et le Traitement du Signal*, Vol. 3 of Editions du Centre National de la Recherche Scientifique, Paris, 1983, pp. 557 – 581.
- [32] A. Isidori, *Nonlinear Control Systems*, Springer Verlag, 1989.
- [33] A. Isidori, Direct construction of minimal bilinear realizations from nonlinear input-output maps, *IEEE Transactions on Automatic Control* (1973) 626–631.
- [34] Y. Wang, E. Sontag, Generating series and nonlinear systems: analytic aspects, local realizability and i/o representations, *Forum Mathematicum* (1992) 299–322.