Reduced-order modeling of LPV systems in the Loewner framework

Ion Victor Gosea* Mihaly Petreczky† Athanasios C. Antoulas‡

*Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany.
Email: gosea@mpi-magdeburg.mpg.de, ORCID: 0000-0003-3580-4116
†Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL), UMR CNRS 9189, CNRS Lille, France.
Email: mihaly.petreczky@ec-lille.fr, ORCID: 0000-0003-2264-5689
‡Rice University, Houston, Texas, USA, and Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany.
Email: aca@rice.edu,

We propose a model reduction method for LPV systems. We consider LPV state-space representations with an affine dependence on the scheduling variables. The main idea behind the proposed method is to compute the reduced order model in such a manner that its frequency domain transfer function coincides with that of the original model for some frequencies. The proposed method uses Loewner-like matrices, which can be calculated from the frequency domain representation of the system. The contribution of the paper represents an extension of the well-established Loewner framework to LPV models.

1 Introduction

Linear parameter-varying (LPV) systems are linear systems where the coefficients are functions of a time-varying signal, the so-called *scheduling variable*. Control design and system identification of LPV systems is a popular topic [1–11]. Model reduction refers to a general class of methodologies used to reduce the complexity of typically large-scale models, by approximating them with simpler, smaller models (and by retaining, at the same time, the main characteristics of the original model). We refer the reader to [12–14], and to the references therein for more details on some of the recent methods developed. Model reduction has also been investigated for LPV systems in the last two decades; we refer the reader to the collection [5, 15–24], for more details. However, model reduction of LPV systems preserving some component of the frequency response has not been investigated so far, to the best of our knowledge.

In this paper we propose a model reduction method which preserves some component of the frequency response of an LPV model. We will concentrate on LPV state-space representations with an affine dependence on the scheduling parameters. This approach is an extension of the well-known Loewner framework for LTI systems [25] and it is closely related to the Loewner framework for linear switched systems [26] and bilinear systems [27]. The basic idea is to define a set of *generalized transfer functions* which represent the multivariate Laplace transforms of the input-output map of an LPV system. The definition of these generalized transfer functions resembles that of bilinear systems [28], and it is closely related to generalized kernel functions for linear switched systems [26]. Similarly, the ensuing Loewner framework formulated here for LPV systems follows closely that for linear switched systems [26], and bears some resemblance with that for bilinear systems [27].

The motivation for formulating a moment matching model reduction algorithm for LPV systems is as follows. First, it allows to deal with LPV systems which are not quadratically stable. This is in contrast to model reduction methods based on balanced truncation or solving LMIs [15–18, 21, 24], and its computation complexity is likely to be lower than that of methods based on solving LMIs. Second, it has a system theoretic interpretation in frequency domain. Finally, in contrast to moment matching methods based on matching sub-Markov parameters [20], the input-output behavior of the reduced model is an approximation of the original one for scheduling signals and control inputs which are linear combinations of certain harmonics. That is, it is possible to relate the frequency response of the original and reduced model. In turn, for LPV control synthesis the use of frequency domain specifications is quite natural, rendering the model reduction method compatible with control design.

To the best of our knowledge, the results of the paper are new. The existing literature is mostly applicable for stable LPV systems. The method of [19] is applicable to quadratically stabilizable and detectable LPV systems. In contrast, this paper does not impose any stability restrictions on the class of LPV systems. In [5], a modification of the realization algorithm is proposed. However, it requires the construction of the Hankel matrix and hence it suffers from the curse of dimensionality. In [29], reduction of the number of states and the number of scheduling parameters was investigated. However, the method of [29] requires constructing the Hankel matrix explicitly. Hence, it displays the same type of challenges as the method in [5].

Outline: In Section 2 we present the definition of the model class, their input-output maps, equivalence and minimality, following [30]. In Section 3, the definition of generalized transfer functions for LPV models is presented. In Section 4 contains a brief introduction to the classical Loewner framework for LTI systems. Section 5 contains the presentation of the main result. In Section 6 we present a numerical example to illustrate the proposed model reduction method.

2 Preliminaries

2.1 Notation and terminology

Let $\mathbb{N}$ be the set of all natural numbers including zero. For a finite set X, denote by $\mathscr{S}(X)$ the set of finite sequences generated by elements from X, *i.e.*, each $s \in \mathscr{S}(X)$ is of the form $s = \zeta_1\zeta_2\cdots\zeta_k$ with $\zeta_1,\zeta_2,\ldots,\zeta_k \in X$, $k \in \mathbb{N}$; $|s|$ denotes the length of the sequence s. For $s,r \in \mathscr{S}(X)$, $sr \in \mathscr{S}(X)$ denotes the concatenation of s and r. The symbol ε is used for the empty sequence and $|\varepsilon| = 0$ with $s\varepsilon = \varepsilon s = s$. Denote by $X^{\mathbb{N}}$ the set of all functions of the form $f : \mathbb{N} \to X$. Let $\mathbb{I}_{\tau_1}^{\tau_2} = \{s \in \mathbb{Z} \mid \tau_1 \leq s \leq \tau_2\}$ be an index set.

Let $\mathbb{T} = \mathbb{R}_0^+ = [0,+\infty)$ be the continuous-time time axis.

A function $f : \mathbb{R}_0^+ \to \mathbb{R}^n$ 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 $\mathscr{C}_\mathrm{p}(\mathbb{R}_0^+,\mathbb{R}^n)$ the set of all *piecewise-continuous functions* of the above form. We denote by $\mathscr{C}_\mathrm{d}(\mathbb{R}_0^+,\mathbb{R}^n)$ the set of all differentiable functions of the form $f : \mathbb{R}_0^+ \to \mathbb{R}^n$.

2.2 System theoretic definitions

An LPV *state-space* (SS) representation with *affine* linear dependence on the *scheduling variable* (abbreviated as LPV-SSA) is a state-space representation of the form

$$\Sigma \left\{ \begin{array}{rcl} \dot{x}(t) & = & A(p(t))x(t) + B(p(t))u(t), \\ y(t) & = & C(p(t))x(t) + D(p(t))u(t), \end{array} \right. \tag{1}$$

where $x(t) \in \mathbb{X} = \mathbb{R}^{n_\mathrm{x}}$ is the state, $y(t) \in \mathbb{Y} = \mathbb{R}^{n_\mathrm{y}}$ is the output, $u(t) \in \mathbb{U} = \mathbb{R}^{n_\mathrm{u}}$is the input, and $p(t) \in \mathbb{P} \subseteq \mathbb{R}^{n_\mathrm{p}}$ is the value of the *scheduling variable* at time t, and A,B,C,D are matrix valued functions on $\mathbb{P}$ defined as

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

for every $\mathbb{p} = [\begin{array}{cccc} \mathbb{p}_1 & \mathbb{p}_2 & \cdots & \mathbb{p}_{n_\mathrm{p}} \end{array}]^\top \in \mathbb{P}$, with constant matrices $A_i \in \mathbb{R}^{n_\mathrm{x}\times n_\mathrm{x}}$, $B_i \in \mathbb{R}^{n_\mathrm{x}\times n_\mathrm{u}}$, $C_i \in \mathbb{R}^{n_\mathrm{y}\times n_\mathrm{x}}$ and $D_i \in \mathbb{R}^{n_\mathrm{y}\times n_\mathrm{u}}$ for all $i \in \mathbb{I}_0^{n_\mathrm{p}}$. *It is assumed that* $\mathbb{P}$ *contains an affine basis of* $\mathbb{R}^{n_\mathrm{p}}$ (see [31] for the definition of an affine basis). In the sequel, we use the tuple

$$\Sigma = (\mathbb{P}, \{A_i, B_i, C_i, D_i\}_{i=0}^{n_\mathrm{p}})$$

to denote an LPV-SSA of the form (1) and use $\dim(\Sigma) = n_\mathrm{x}$ to denote its state dimension. Define $\mathscr{X} = \mathscr{C}_\mathrm{d}(\mathbb{R}_0^+,\mathbb{X})$, $\mathscr{Y} = \mathscr{C}_\mathrm{p}(\mathbb{R}_0^+,\mathbb{Y})$, $\mathscr{U} = \mathscr{C}_\mathrm{p}(\mathbb{R}_0^+,\mathbb{U})$, $\mathscr{P} = \mathscr{C}_\mathrm{p}(\mathbb{R}_0^+,\mathbb{P})$. By a solution of Σ we mean a tuple of trajectories $(x,y,u,p) \in (\mathscr{X},\mathscr{Y},\mathscr{U},\mathscr{P})$ such that (1) holds for all $t \in \mathbb{T}$. For an initial state $x_\mathrm{o} \in \mathbb{X}$ define the *input-to-state map* $\mathfrak{X}_{\Sigma,x_\mathrm{o}}$ and the *input-output* map $\mathfrak{Y}_{\Sigma,x_\mathrm{o}}$ of Σ induced by x_o as

$$\mathfrak{X}_{\Sigma,x_\mathrm{o}} : \mathscr{U} \times \mathscr{P} \to \mathscr{X}, \quad \mathfrak{Y}_{\Sigma,x_\mathrm{o}} : \mathscr{U} \times \mathscr{P} \to \mathscr{Y}, \tag{3}$$

such that for any $(x,y,u,p) \in \mathscr{X} \times \mathscr{Y} \times \mathscr{U} \times \mathscr{P}$, $x = \mathfrak{X}_{\Sigma,x_\mathrm{o}}(u,p)$ and $y = \mathfrak{Y}_{\Sigma,x_\mathrm{o}}(u,p)$ holds if and only if (x,y,u,p) is a solution of (1) and $x(0) = x_\mathrm{o}$.

We say that Σ is *span-reachable* from an initial state $x_\mathrm{o} \in \mathbb{X}$, if $\mathrm{Span}\{\mathfrak{X}_{\Sigma,x_\mathrm{o}}(u,p)(t) \mid (u,p) \in \mathscr{U} \times \mathscr{P}, t \in \mathbb{T}\} = \mathbb{X}$. In this paper we will concentrate on zero initial states, hence we will say that Σ is *span-reachable*, if it is span-reachable from the zero initial state. We say that Σ is *observable*, if for any two initial states $\bar{x}_\mathrm{o}, \hat{x}_\mathrm{o} \in \mathbb{R}^{n_\mathrm{x}}$, $\mathfrak{Y}_{\Sigma,\hat{x}_\mathrm{o}} = \mathfrak{Y}_{\Sigma,\bar{x}_\mathrm{o}}$ implies $\hat{x}_\mathrm{o} = \bar{x}_\mathrm{o}$. Let Σ of the form (1) and $\Sigma' = (\mathbb{P}, \{A_i', B_i', C_i', D_i'\}_{i=0}^{n_\mathrm{p}})$ with $\dim(\Sigma) = \dim(\Sigma') = n_\mathrm{x}$. A nonsingular matrix $T \in \mathbb{R}^{n_\mathrm{x}\times n_\mathrm{x}}$ is said to be an *isomorphism* from Σ to Σ', if

$$\forall i \in \mathbb{I}_0^{n_\mathrm{p}} : A_i'T = TA_i,\ B_i' = TB_i,\ C_i'T = C_i,\ D_i' = D_i.$$

We formalize the input-output behavior of LPV-SSAs as maps of the form

$$\mathfrak{F} : \mathscr{U} \times \mathscr{P} \to \mathscr{Y}. \tag{4}$$

While any input-output map of an LPV-SSA induced by some initial state is of the above form, the converse is not true. The LPV-SSA Σ is a *realization* of an input-output map $\mathfrak{F}$ of the form (4) from *the initial state* $x_\mathrm{o} \in \mathbb{X}$, if $\mathfrak{F} = \mathfrak{Y}_{\Sigma,x_\mathrm{o}}$. In this paper we will concentrate on LPV-SSA realizations from the zero initial state. Accordingly, we will say Σ is *realization* of $\mathfrak{F}$, if Σ is a realization of $\mathfrak{F}$ from zero initial state. An LPV-SSA Σ is a *minimal* *realization of* $\mathfrak{F}$ *from the initial state* x_{o}, if Σ is a realization of $\mathfrak{F}$ from the initial state x_{o}, and for every LPV-SSA Σ' which is a realization of $\mathfrak{F}$, $\dim(\Sigma) \leq \dim(\Sigma')$. Again, when the initial state is zero, we say that Σ is a *minimal realization* of $\mathfrak{F}$, if Σ is a minimal realization of $\mathfrak{F}$ from the zero initial state. It can be shown that a LPV-SSA is a minimal realization of an input-output map, if and only if it is span-reachable and observable, moreover, all minimal LPV-SSA realizations of the same input-output map are isomorphic [30]. Furthermore, span-reachability and observability can be characterized by rank conditions of suitably defined matrices [30].

In this paper, in order to avoid excessive notation, we will make the following simplifying assumptions on the LPV-SSA models considered.

Assumption 1 *In the sequel we assume that there is only one control input, i.e.,* $n_{\mathrm{u}} = 1$ *and we consider only LPV-SSA models of the form* (1) *for which the* D *matrix is zero, and* C *and* B *matrices do not depend on the scheduling parameters, i.e.,* $C(p) = C_0, B(p) = B_0$, $D(p) = 0$ *and hence* $D_0 = 0$, $C_i = 0, B_i = 0, D_i = 0$ *for all* $i = 1, \dots, n_{\mathrm{p}}$,

3 Generalized transfer functions for LPV-SSA

Note that an input-output map $\mathfrak{F}$ of the form (4) is realizable by an LPV-SSA from the zero initial state satisfying Assumption 1, only if $\mathfrak{F}$ admits a so called impulse response representation [30], i.e., only if

$$\mathfrak{F}(u,p)(t) = \int_0^t (h_{\mathfrak{F}} \diamond p)(\delta,t)u(\delta)\, d\delta, \tag{5}$$

where for every $p \in \mathscr{P}$ the function $(h_{\mathfrak{F}} \diamond p)$ satisfies a number of technical conditions. These conditions imply that $(h_{\mathfrak{F}} \diamond p)$ is an input-output map induced by generating series in the sense of [32], where the scheduling signal p plays the role of the input. Recall from [32, Chapter 3, Section 3.2] that input-output maps which are induced by generating series also admit a Volterra-series representation. Moreover, if $\mathfrak{F}$ has a realization by a LPV-SSA, then the input-output map w$p \mapsto (h_{\mathfrak{F}} \diamond p)$ can be realized by a bilinear system whose matrices are matrices of the LPV-SSA realization of $\mathfrak{F}$. More precisely, by [30] there exists a generating (Fliess) series $\theta_{\mathfrak{F}} : \mathfrak{S}(\mathbb{I}_0^{n_{\mathrm{p}}}) \to \mathbb{R}^{n_{\mathrm{y}}}$ defined on the set of all sequence of elements of $\mathbb{I}_0^{n_{\mathrm{p}}} = \{0, 1, \dots, n_{\mathrm{p}}\}$, such that

$$\begin{aligned}
&(h \diamond p)(\delta,t) = F_{\theta_{\mathfrak{F}}}[\sigma_\delta p](t-\delta) = \\
&\theta_{\mathfrak{F}}(\varepsilon) + \sum_{k=1}^{\infty} \sum_{i_1,\dots,i_k=0}^{n_{\mathrm{p}}} \theta_{\mathfrak{F}}(i_1 \cdots i_k) \times \\
&\times \int_\delta^t \int_\delta^{\tau_k} \cdots \int_\delta^{\tau_2} p_{i_k}(\tau_k) \cdots p_{i_1}(\tau_1) d\tau_k \cdots d\tau_1.
\end{aligned} \tag{6}$$

Here, $p_0 = 1$, $F_{\theta_{\mathfrak{F}}}$ denotes the input-output map induced by the generating series $\theta_{\mathfrak{F}}$, $\sigma_\delta p : s \mapsto p(\delta + s)$, and $F_{\theta_{\mathfrak{F}}}[\tilde{p}]$ is the value of the input-output map $F_{\theta_{\mathfrak{F}}}$ for the input signal $\tilde{p}$. Here we use the standard notation used for generating (Fliess) series, see [32]. The second equation in (6) is just the definition of an input-output map induced by a generating series. If Σ is of the form (1), satisfying Assumption 1, with $B(p) = B_0 = B$ and $C(p) = C_0 = C$, and Σ is a realization of $\mathfrak{F}$, then it holds that

$$\theta_{\mathfrak{F}}(s) = CA_sB, \tag{7}$$

where for $s{=}\varepsilon$, A_s is the identity matrix, and for $s{=}s_1s_2\cdots s_n$, $s_1, \dots, s_n \in \{0, \dots n_{\mathrm{p}}\}$, $n{>}0$, then $A_s = A_{s_n}A_{s_{n-1}}\cdots A_{s_1}$.

In other words, , the input-output map $F_{\theta_{\mathfrak{F}}}$ is the input-output map of the bilinear system

$$\begin{aligned}
\dot{z}(t) &= (A_0 + \sum_{i=1}^{n_{\mathrm{p}}} A_i p_i(t))z(t),\ z(0) = B, \\
y(t) &= Cz(t),
\end{aligned} \tag{8}$$

and hence, $y(t) = h_{\mathfrak{F}} \diamond p(\delta,t)$ is the output of the following bilinear system at time t

$$\begin{aligned}
\dot{z}(t) &= (A_0 + \sum_{i=1}^{n_{\mathrm{p}}} A_i p_i(t))z(t),\ z(\delta) = B, \\
y(t) &= Cz(t),
\end{aligned} \tag{9}$$

driven by the scheduling signal interpreted as input.

Recall from [32, Chapter 3, Section 3.2] that input-output maps induced by generating series can also be represented by Volterra-kernels. For the specific case of $h_{\mathfrak{F}} \diamond p$, this representation is as presented here; let us define functions $W^{\mathfrak{F}}_{q_1,\dots,q_k}(\tau_k, \tau_k, \dots, \tau_0)$, $\tau_k \geq \tau_{k-1} \geq \cdots \geq \tau_0 \geq 0$, $q_1, \dots, q_k \in \{1, \dots, n_{\mathrm{p}}\}$ and $W^{\mathfrak{F}}_0(\tau_0)$ as follows

$$\begin{aligned}
&W^{\mathfrak{F}}_{q_1\cdots q_k}(\tau_k, \tau_k, \dots, \tau_0) = \\
&\sum_{n_0,\dots,n_k \in \mathbb{N}} \theta_{\mathfrak{F}}(0^{n_0}q_1 0^{n_1} \cdots q_k 0^{n_k}) \frac{\tau_0^{n_0}}{n_0!} \Pi_{i=1}^k \frac{(\tau_i - \tau_{i-1})^{n_i}}{n_i!} \\
&W^{\mathfrak{F}}_0(\tau_0) = \sum_{n_0 \in \mathbb{N}} \theta_{i,j,\mathfrak{F},l}(0^{n_0}) \frac{\tau^{n_0}}{n_0!}.
\end{aligned} \tag{10}$$

Here 0^k represents the k-fold repetition of the symbol 0, i.e., $0^0 = \varepsilon$, $0^k = \underbrace{00\cdots0}_{k\text{times}}$. It then follows that

$$\begin{aligned}
&(h_{\mathfrak{F}} \diamond p)(\delta,t) = W^{\mathfrak{F}}_0(t-\delta) + \\
&\sum_{k=1}^{\infty} \sum_{q_1\dots q_k=1}^{n_{\mathrm{p}}} \int_\delta^t \int_\delta^{\tau_k} \cdots \int_\delta^{\tau_1} W^{\mathfrak{F}}_{q_1,\dots,q_k}(t-\delta, \tau_k-\delta, \dots, \tau_1-\delta) \times \\
&p_{q_k}(\tau_k) \cdots p_{q_1}(\tau_1) d\tau_k \cdots d\tau_1 = \\
&W^{\mathfrak{F}}_0(t-\delta) + \\
&\sum_{k=1}^{\infty} \sum_{q_1\dots q_k=1}^{n_{\mathrm{p}}} \int_0^{t-\delta} \int_0^{\tau_k} \cdots \int_0^{\tau_1} W^{\mathfrak{F}}_{q_1,\dots,q_k}(t-\delta, \tau_k, \dots, \tau_1) \times \\
&p_{q_k}(\tau_k+\delta) \cdots p_{q_1}(\tau_1+\delta) d\tau_k \cdots d\tau_1.
\end{aligned}$$

In particular, if Σ is a realization of $\mathfrak{F}$ of the form (1), with $C(p) = C$ and $B(p) = B$ being constants, then

$$W_0^{\mathfrak{F}}(t) = Ce^{A_0 t}B,$$
$$W_{q_1,\ldots,q_k}^{\mathfrak{F}}(\tau_k,\tau_{k-1},\ldots,\tau_0) =$$
$$Ce^{A_0(\tau_k-\tau_{k-1})}A_{q_k}e^{A_0(\tau_{k-1}-\tau_{k-2})}A_{q_{k-1}}\cdots e^{A_0(\tau_1-\tau_0)}A_{q_1}e^{A_0\tau_0}B$$

The Volterra-kernels (10) are the classical Volterra-kernels of input-affine nonlinear systems. In particular, we can take their multivariate Laplace transforms resulting in a sequence of transfer functions $H_0(s)^{\mathfrak{F}}$, $H_{q_1,\ldots,q_k}^{\mathfrak{F}}(s_0,s_1,\ldots,s_k)$

$$H_{q_1\cdots q_k}^{\mathfrak{F}}(s_0,s_1,s_2\ldots,s_k) =$$
$$\int_0^\infty\cdots\int_0^\infty W^{\mathfrak{F}}(\sum_{j=0}^{k}\tau_j,\sum_{j=0}^{k-1}\tau_j,\ldots,,\tau_0)e^{-(\Sigma_{j=1}^k s_j\tau_j)}d\tau_0\cdots d\tau_k,$$
$$H_0^{\mathfrak{F}}(s) = \int_0^\infty W_0^{\mathfrak{F}}(\tau)e^{-s\tau}d\tau. \tag{11}$$

Strictly speaking, the right-hand sides of in the equations (11) are well-defined only if $\Re(s_i) > \sigma$ for a suitably chosen real number σ which depends on k and $q_1,\ldots,q_k$. In particular, if A_0 is stable, then σ above can be taken to be 0. For the sake of simplicity, in the sequel we will implicitly assume that the functions $H_0^{\mathfrak{F}}$ and $H_{q_1,\ldots,q_k}^{\mathfrak{F}}$ are evaluated only for arguments for which the right-hand side of (11) is convergent.

In Σ is a LPV-SSA realization of $\mathfrak{F}$ of the form (1) with $C(p) = C$ and $B(p) = B$, then

$$\begin{aligned}&H_{q_1\cdots q_k}^{\mathfrak{F}}(s_0,s_1,s_2\ldots,s_k) =\\ &C\Phi(s_k)A_{q_k}\Phi(s_{k-1})A_{q_{k-1}}\cdots A_{q_2}\Phi(s_1)A_{q_1}\Phi(s_0)B,\\ &H_0^{\mathfrak{F}}(s_0) = C\Phi(s_0)B,\end{aligned} \tag{12}$$

where $\Phi(s) = (sI - A_0)^{-1}$ for all $s \in \mathbb{C}$.

Definition 1 (Generalized transfer functions) *The following sequence of transfer functions given by*

$$\{H_0^{\mathfrak{F}}, H_{q_1\cdots q_k}^{\mathfrak{F}} \mid q_1,\ldots,q_k \in \{1,\ldots,n_\mathrm{p}\}, k > 0\}, \tag{13}$$

is called the sequence of generalized transfer functions of $\mathfrak{F}$.

4 The Loewner framework for modeling classical LTI systems

In this section we present a brief overview of the Loewner framework, originally introduced in [25], for the LTI systems with multiple inputs and multiple outputs. For more details on various aspects of the method, we refer the reader to [33]. This framework is a data-driven modeling approach that constructs an LTI dynamical model with transfer function $H_M : \mathbb{C} \to \mathbb{C}^{p\times m}$ which interpolates the given $2M$ samples (data measurements), for $M \in \mathbb{N}_+$. Let the left (or row) data values be given together with the right (or column) data values, as follows

$$\left.\begin{array}{c}(\mu_j, l_j^T, v_j^T)\\ \text{for } j = 1,\ldots,M\end{array}\right\} \text{ and } \left\{\begin{array}{c}(\lambda_i, r_i, w_i)\\ \text{for } i = 1,\ldots,M\end{array}\right., \tag{14}$$

where $v_j^T = l_j^T H(\mu_j)$ and $w_i = H(\lambda_i)r_i$, with $l_j \in \mathbb{C}^{p\times 1}$, $r_i \in \mathbb{C}^{m\times 1}$, $v_j \in \mathbb{C}^{m\times 1}$ and $w_i \in \mathbb{C}^{p\times 1}$. Then, split the distinct interpolation points $\{\eta_k\}_{k=1}^{2M} \subset \mathbb{C}$ is split up into two disjoint subsets of same size, *i.e.*

$$\{\eta_k\}_{k=1}^{2M} = \{\mu_j\}_{j=1}^M \cup \{\lambda_i\}_{i=1}^M. \tag{15}$$

The first step is to compute two matrices, i.e., the *Loewner* matrix $\mathbb{L} \in \mathbb{C}^{M\times M}$ and *shifted Loewner* matrix $\mathbb{L}_s \in \mathbb{C}^{M\times M}$ defined for $i = 1,\ldots,M$ and $j = 1,\ldots,M$, as:

$$\begin{aligned}[\mathbb{L}]_{j,i} &= \frac{v_j^T r_i - l_j^T w_i}{\mu_j - \lambda_i} = \frac{l_j^T\left(H(\mu_j) - H(\lambda_i)\right)r_i}{\mu_j - \lambda_i},\\ [\mathbb{L}_s]_{j,i} &= \frac{\mu_j v_j^T r_i - \lambda_i l_j^T w_i}{\mu_j - \lambda_i} = \frac{l_j^T\left(\mu_j H(\mu_j) - \lambda_i H(\lambda_i)\right)r_i}{\mu_j - \lambda_i}.\end{aligned} \tag{16}$$

Additionally, we introduce the following matrices

$$\mathbb{V} = \begin{bmatrix}v_1 & \cdots & v_M\end{bmatrix}^T, \quad \mathbb{W} = \begin{bmatrix}w_1 & \cdots & w_M\end{bmatrix}, \tag{17}$$

with the following notation that holds for all $j,i = 1,\ldots,M$

$$v_j^T = l_j^T H(\mu_j), \quad \text{and} \quad w_i = H(\lambda_i)r_i. \tag{18}$$

Then, the Loewner LTI model Σ_M is characterized by the following realization,

$$\Sigma_M : \begin{cases}E_M\dot{\mathbf{x}}(t) = A_M\mathbf{x}(t) + B_M\mathbf{u}(t),\\ \mathbf{y}(t) = C_M\mathbf{x}(t),\end{cases} \tag{19}$$

where $E_M = -\mathbb{L}$, $A_M = -\mathbb{L}_s$, $B_M = \mathbb{V}$ and $C_M = \mathbb{W}$. The transfer function of Σ_M is given by

$$H_M(s) = C_M(sE_M - A_M)^{-1}B_M. \tag{20}$$

Theorem 1 *Given the framework previously introduced, the function H_M interpolates H at the given driving frequencies and directions, i.e., for all $1 \le i \le M$, it holds that*

$$\begin{aligned}l_j^H H_M(\mu_j) &= l_j^H H(\mu_j),\\ H_M(\lambda_i)r_i &= H(\lambda_i)r_i.\end{aligned} \tag{21}$$

Next, we assume that the number of available measurements is larger than the underlying system's order denoted with n, i.e., $2M \ge n$. In this case, it was shown in [25] that a minimal model H_n of dimension $n < M$ (that still interpolates the data) can be computed by means of projecting (19). In order for this to be possible, the conditions below

$$\mathrm{rank}(\eta_k \mathbb{L} - \mathbb{L}_s) = \mathrm{rank}([\mathbb{L}, \mathbb{L}_s]) = \mathrm{rank}([\mathbb{L}^T, \mathbb{L}_s{}^T]^T) = n, \quad (22)$$

need to hold for $k = 1, \dots, 2M$, where η_k are as in (15). In that case, let $Y \in \mathbb{C}^{M \times n}$ be the matrix containing the first n left singular vectors of $[\mathbb{L}, \mathbb{L}_s]$ and $X \in \mathbb{C}^{M \times n}$ the matrix containing the first n right singular vectors of $[\mathbb{L}^T, \mathbb{L}_s{}^T]^T$. Then, construct a realization by means of projection as

$$\begin{aligned} E_n &= Y^T E_M X, A_n = Y^T A_M X, \\ B_n &= Y^T B_M, C_n = C_M X, \end{aligned} \quad (23)$$

which is equivalent to that in (19). The realization in (23) encodes a *minimal McMillan degree* equal to $\mathrm{rank}(\mathbb{L})$.

Finally, the number of singular vectors (n) that enter matrices Y and X in (23) could be indeed decreased to a value $r < n$. This would result in computing a reduced r-th order rational model that approximately interpolates the data. This allows a trade-off between complexity of the resulting model and accuracy of interpolation (as explained in [25]).

5 The proposed procedure

In what follows we describe the proposed procedure to construct a reduced order LPV-SSA $\tilde{\Sigma} = (\mathbb{P}, \{\tilde{A}_i, \tilde{B}_i, \tilde{C}_i, \tilde{D}_i\}_{i=0}^{n_p})$ from an LPV-SSA of the form (1) satisfying Assumption 1.

To this end, let $N \in \mathbb{N}$ be a positive integer and introduce the following sequences of scalars:

1. $(\mu_0, \mu_1, \dots, \mu_N)$ is the tuple of *left interpolation points* in the frequency domain, with $\mu_i \in \mathbb{C}$.
2. $(\lambda_0, \lambda_1, \dots, \lambda_N)$ is the tuple of *right interpolation points* in the frequency domain with $\lambda_j \in \mathbb{C}$.
3. $(q_1^{(l)} q_2^{(l)} \dots q_N^{(l)})$ is the word of *left expansion points* in the parameter domain with $q_i^{(l)} \in \mathbb{N}$.
4. $(q_1^{(r)} q_2^{(r)} \dots q_N^{(r)})$ is the word of *right expansion points* in the parameter domain with $q_j^{(r)} \in \mathbb{N}$.

The associated generalized observability matrix $\mathscr{O} \in \mathbb{C}^{(N+1) \times n_x}$ of the LPV-SSA (1) is put together as follows

$$\mathscr{O} = \begin{bmatrix} C\Phi(\mu_0) \\ C\Phi(\mu_0) A_{q_1^{(l)}} \Phi(\mu_1) \\ C\Phi(\mu_0) A_{q_1^{(l)}} \Phi(\mu_1) A_{q_2^{(l)}} \Phi(\mu_2) \\ \vdots \\ C\Phi(\mu_0) A_{q_1^{(l)}} \Phi(\mu_1) A_{q_2^{(l)}} \Phi(\mu_2) \cdots A_{q_N^{(l)}} \Phi(\mu_N) \end{bmatrix}. \quad (24)$$

Recall that $C = C_0$, as by Assumption 1 C does not depend on the scheduling variable. Additionally, the associated generalized controllability matrix $\mathscr{R} \in \mathbb{C}^{n_x \times (N+1)}$ of (1) is also put together; below, we explicitly provide the entries of the jth column of matrix $\mathscr{R}$, denote with $\mathscr{R}_j$, as

$$\begin{aligned} &\mathscr{R}_1 = \begin{bmatrix} \Phi(\lambda_0) B \end{bmatrix}, \ \mathscr{R}_2 = \begin{bmatrix} \Phi(\lambda_1) A_{q_1^{(r)}} \Phi(\lambda_0) B \end{bmatrix} \\ &\mathscr{R}_3 = \begin{bmatrix} \Phi(\lambda_2) A_{q_2^{(r)}} \Phi(\lambda_1) A_{q_1^{(r)}} \Phi(\lambda_0) B \end{bmatrix} \\ &\mathscr{R}_N = \begin{bmatrix} \Phi(\lambda_N) A_{q_N^{(r)}} \Phi(\lambda_{N-1}) A_{q_{N-1}^{(r)}} \Phi(\lambda_{N-2}) \cdots A_{q_1^{(r)}} \Phi(\lambda_0) B \end{bmatrix}. \end{aligned} \quad (25)$$

Recall that $B = N_0$, as by Assumption 1 C does not depend on the scheduling variable. Then, put together a reduced-order model $\tilde{\Sigma}$ for the system in (1), which is constructed from the original quantities. Define matrices for $1 \leq i \leq n_p$

$$\hat{E} = \mathscr{O}\mathscr{R}, \ \hat{A}_0 = \mathscr{O} A_0 \mathscr{R}, \ \hat{A}_i = \mathscr{O} A_i \mathscr{R}, \ \hat{B} = \mathscr{O} B, \ \hat{C} = C\mathscr{R}. \quad (26)$$

Provided that $\hat{E}$ is nonsingular, one can write for all $1 \leq i \leq n_p$:

$$\tilde{E} = I, \ \tilde{A}_0 = \hat{E}^{-1}\hat{A}_0, \ \tilde{A}_i = \hat{E}^{-1}\hat{A}_i, \ \tilde{B} = \hat{E}^{-1}\hat{B}, \ \tilde{C} = \hat{C}. \quad (27)$$

We then define the reduced order model $\tilde{\Sigma}$ as

$$\tilde{\Sigma} = (\mathbb{P}, \{\tilde{A}_i, \tilde{B}_i, \tilde{C}_i, \tilde{D}_i\}_{i=0}^{n_p}), \quad (28)$$

where $\tilde{D}_i = 0$, $i = 0, \dots, n_p$ and $\tilde{C}_i = 0$, $\tilde{B}_i = 0$ for $i = 1, \dots, n_p$ and $\tilde{B}_0 = \tilde{B}, \tilde{C} = \tilde{C}_0$ and $\{\tilde{A}_i\}_{i=1}^{n_p}, \tilde{C}, \tilde{B}$ are as in (27).

5.1 Data-driven interpretation

We will show in this section that the matrices computed in (26) can indeed be expressed in terms of samples of the transfer functions introduced in (11).

For example, one can directly write the entries of vectors $\hat{B} = \mathscr{O}B$ and $\hat{C} = C\mathscr{R}$ in (26) as

$$\hat{B} = \begin{bmatrix} H_0^{\mathfrak{F}}(\mu_0) \\ H_{q_1^{(l)}}^{\mathfrak{F}}(\mu_1, \mu_0) \\ H_{q_2^{(l)}, q_1^{(l)}}^{\mathfrak{F}}(\mu_2, \mu_1, \mu_0) \\ \vdots \end{bmatrix}, \ \hat{C} = \begin{bmatrix} H_0^{\mathfrak{F}}(\lambda_0) \\ H_{q_1^{(r)}}^{\mathfrak{F}}(\lambda_0, \lambda_1) \\ H_{q_1^{(r)}, q_2^{(r)}}^{\mathfrak{F}}(\lambda_0, \lambda_1, \lambda_2) \\ \vdots \end{bmatrix}^T \quad (29)$$

Additionally, the matrices A_i for $1 \leq i \leq n_p$ are written element-wise, as follows:

$$\begin{aligned} (\hat{A}_i)_{k+1, \ell+1} &= \mathscr{O}_{k+1} A_i \mathscr{R}_{\ell+1} \\ &= H^{\mathfrak{F}}_{q_1^{(r)}, \dots, q_\ell^{(r)}, i, q_k^{(l)}, \dots, q_1^{(l)}}(\lambda_0, \dots, \lambda_\ell, \mu_k, \dots, \mu_0) \end{aligned} \quad (30)$$

Next, proceed to explicitly writing the $(k+1,\ell+1)$ entry of matrix $\hat{E}$ for $k,\ell \geq 0$. We make use of the recursion formulas on the rows and columns of matrices $\mathscr{O}$, and respectively, $\mathscr{R}$ as (to have consistent notations, we enforce $\mathscr{O}_0 = \mathscr{R}_0 = 1$):

$$\mathscr{O}_{k+1} = \mathscr{O}_k A_{q_k^{(l)}} \Phi(\mu_k), \;\; \mathscr{R}_{\ell+1} = \Phi(\lambda_\ell) A_{q_\ell^{(l)}} \mathscr{R}_\ell. \tag{31}$$

Hence, based on the two identities presented above, we write the $(k+1,\ell+1)$ entry of matrix $\hat{E}$, for all $0 \leq k,\ell \leq N$, in the following way:

$$\big(\hat{E}\big)_{k+1,\ell+1} = \mathscr{O}_{k+1} I \mathscr{R}_{\ell+1} = \mathscr{O}_k A_{q_k^{(l)}} \Phi(\mu_k) I \Phi(\lambda_\ell) A_{q_\ell^{(r)}} \mathscr{R}_\ell. \tag{32}$$

Next, we make use of the identity: $I = \frac{\Phi^{-1}(\mu_k)-\Phi^{-1}(\lambda_\ell)}{\mu_k-\lambda_\ell}$ and by substituting it in the equality above, it follows that:

$$\begin{aligned}\big(\hat{E}\big)_{k+1,\ell+1} &= \mathscr{O}_{k+1} I \mathscr{R}_{\ell+1} \\ &= \mathscr{O}_k A_{q_k^{(l)}} \Phi(\mu_k) \frac{\Phi^{-1}(\mu_k)-\Phi^{-1}(\lambda_\ell)}{\mu_k-\lambda_\ell} \Phi(\lambda_\ell) A_{q_\ell^{(r)}} \mathscr{R}_\ell \\ &= -\frac{\alpha_{k,\ell}-\beta_{k,\ell}}{\mu_k-\lambda_\ell},\end{aligned} \tag{33}$$

where $\forall 0 \leq k,\ell \leq N,$ and the following notations are used:

$$\begin{aligned}\alpha_{k,\ell} &= \mathscr{O}_k A_{q_k^{(l)}} \Phi(\mu_k) A_{q_\ell^{(r)}} \mathscr{R}_\ell \\ &= H^{\mathfrak{F}}_{q_1^{(r)},\ldots,q_\ell^{(r)},q_k^{(l)},\ldots,q_1^{(l)}}(\lambda_0,\ldots,\lambda_{\ell-1},\mu_k,\ldots,\mu_0), \\ \beta_{k,\ell} &= \mathscr{O}_k A_{q_k^{(l)}} \Phi(\lambda_\ell) A_{q_\ell^{(r)}} \mathscr{R}_\ell \\ &= H^{\mathfrak{F}}_{q_1^{(r)},\ldots,q_\ell^{(r)},q_k^{(l)},\ldots,q_1^{(l)}}(\lambda_0,\ldots,\lambda_\ell,\mu_{k-1},\ldots,\mu_0).\end{aligned} \tag{34}$$

Hence, we have shown that the entries of matrix $\hat{E}$ are divided differences composed of measurements corresponding to transfer functions in (11). We proceed similarly for the entries of matrix $\hat{A}_0$. By using identity $A_0 = \frac{\mu_k \Phi^{-1}(\mu_k)-\lambda_\ell \Phi^{-1}(\lambda_\ell)}{\mu_k-\lambda_\ell}$, it follows

$$\big(\hat{A}_0\big)_{k,\ell} = \mathscr{O}_k A_0 \mathscr{R}_\ell = -\frac{\mu_k \alpha_{k,\ell} - \lambda_\ell \beta_{k,\ell}}{\mu_k-\lambda_\ell}, \tag{35}$$

where α_k and β_ℓ are as defined in (34), i.e., as samples of transfer functions introduced in (11). So, we have shown that all matrices forming the data-driven surrogate realization in (26) are composed of transfer function measurements.

Remark 1 *The matrix $\hat{E} \in \mathbb{C}^{(N+1)\times(N+1)}$ defined element-wise as in (33) is a Loewner matrix, while the matrix $\hat{A}_0 \in \mathbb{C}^{(N+1)\times(N+1)}$ in (35) is a "shifted Loewner matrix", by following the terminology introduced in [25].*

5.2 Interpolation property

In this section we will show that the reduced model satisfies interpolation conditions.

For the reduced-order LPV-SSA $\tilde{\Sigma}$ given in (28), let $\tilde{\mathfrak{F}}$ be the input-output map of $\tilde{\Sigma}$. It then follows that

$$\begin{aligned}&H^{\tilde{\mathfrak{F}}}_{q_1\cdots q_k}(s_1,s_2\ldots,s_k) = \\ &\tilde{C}\widetilde{\Phi}(s_k)\tilde{A}_{q_k}\widetilde{\Phi}(s_{k-1})A_{q_{k-1}}\cdots\tilde{A}_{q_2}\widetilde{\Phi}(s_1)A_{q_1}\widetilde{\Phi}(s_0)B, \\ &H^{\tilde{\mathfrak{F}}}_0(s_0) = \tilde{C}\widetilde{\Phi}(s_0)\tilde{B},\end{aligned} \tag{36}$$

where $\widetilde{\Phi}(s) = (sI-\tilde{A}_0)^{-1}$ for all $s \in \mathbb{C}$.

Given unit vectors $\mathbf{e}_{k+1}, \mathbf{e}_1 \in \mathbb{R}^{N+1}$, one can write that:

$$\begin{aligned}\mathbf{e}_{k+1}^T \hat{E}\widetilde{\Phi}^{-1}(\lambda_0)\mathbf{e}_1 &= \mathbf{e}_{k+1}^T \hat{E}(\lambda_0 I - \tilde{A}_0)\mathbf{e}_1 \\ &= \mathbf{e}_{k+1}^T(\lambda_0\hat{E}-\hat{A}_0)\mathbf{e}_1 \\ &= \lambda_0\mathbf{e}_{k+1}^T\hat{E}\mathbf{e}_1 - \mathbf{e}_{k+1}^T\hat{A}_0\mathbf{e}_1 \\ &= -\lambda_0\frac{\alpha_{k,0}-\beta_{k,0}}{\mu_k-\lambda_0} + \frac{\mu_k\alpha_{k,0}-\lambda_0\beta_{k,0}}{\mu_k-\lambda_0} \\ &= \alpha_{k,0} = \mathbf{e}_{k+1}^T\hat{B} = \mathbf{e}_{k+1}^T\hat{E}\tilde{B}\end{aligned} \tag{37}$$

Hence, we have shown that $\mathbf{e}_{k+1}^T \hat{E}\widetilde{\Phi}^{-1}(\lambda_0)\mathbf{e}_1 = \mathbf{e}_{k+1}^T\hat{E}\tilde{B}$, $\forall 0 \leq k \leq N$ which implies that $\widetilde{\Phi}^{-1}(\lambda_0)\mathbf{e}_1 = \tilde{B}$. By multiplying this identity to the left with $\tilde{C}\widetilde{\Phi}(\lambda_0)$, we can write that:

$$\begin{aligned}&\tilde{C}\widetilde{\Phi}(\lambda_0)\widetilde{\Phi}^{-1}(\lambda_0)\mathbf{e}_1 = \tilde{C}\widetilde{\Phi}(\lambda_0)\tilde{B} \Rightarrow \tilde{C}\mathbf{e}_1 = \tilde{C}\widetilde{\Phi}(\lambda_0)\tilde{B} \\ &\Rightarrow H^{\tilde{\mathfrak{F}}}_0(\lambda_0) = H^{\mathfrak{F}}_0(\lambda_0).\end{aligned} \tag{38}$$

Here we used that $\tilde{C} = \hat{C} = C\mathscr{R}$ and hence it follows that $\tilde{C}\mathbf{e}_1 = C\mathscr{R}\mathbf{e}_1 = C\Phi(\lambda_0)B = H^{\mathfrak{F}}_0(\lambda_0)$, where $\Phi(s) = (sI-A_0)^{-1}$. By repeating the above procedure, we can show that the interpolation condition $\hat{H}^{\mathfrak{F}}_0(\lambda_0) = H^{\mathfrak{F}}_0(\lambda_0)$ also holds. In general, we can show that all measurements that appear as entries in the matrices of the reduced-order realization (28), are actually matched by (36). More precisely, we formulate the following result that explicitly states the interpolation conditions satisfied by the surrogate model.

Theorem 2 *Given the framework previously introduced, the following $(N+1)^2 + n_{\rm p}(N+1)^2$ interpolation conditions are satis-*

fied by the transfer functions in (36):

$$
\begin{aligned}
&H^{\tilde{\mathfrak{F}}}_{q_1^{(r)},\dots,q_\ell^{(r)},q_k^{(l)},\dots,q_1^{(l)}}(\lambda_0,\dots,\lambda_\ell,\mu_{k-1},\dots,\mu_0)\\
&= H^{\mathfrak{F}}_{q_1^{(r)},\dots,q_\ell^{(r)},q_k^{(l)},\dots,q_1^{(l)}}(\lambda_0,\dots,\lambda_\ell,\mu_{k-1},\dots,\mu_0),\\
&H^{\tilde{\mathfrak{F}}}_{q_1^{(r)},\dots,q_\ell^{(r)},q_k^{(l)},\dots,q_1^{(l)}}(\lambda_0,\dots,\lambda_{\ell-1},\mu_k,\dots,\mu_0)\\
&= H^{\mathfrak{F}}_{q_1^{(r)},\dots,q_\ell^{(r)},q_k^{(l)},\dots,q_1^{(l)}}(\lambda_0,\dots,\lambda_{\ell-1},\mu_k,\dots,\mu_0),\\
&H^{\tilde{\mathfrak{F}}}_{q_1^{(r)},\dots,q_\ell^{(r)},i,q_k^{(l)},\dots,q_1^{(l)}}(\lambda_0,\dots,\lambda_\ell,\mu_k,\dots,\mu_0)\\
&= H^{\mathfrak{F}}_{q_1^{(r)},\dots,q_\ell^{(r)},i,q_k^{(l)},\dots,q_1^{(l)}}(\lambda_0,\dots,\lambda_\ell,\mu_k,\dots,\mu_0),
\end{aligned}
\tag{39}
$$

for all $0 \le k,\ell \le N$ *and* $1 \le i \le n_{\mathrm{p}}$.

Example 1 *Below, we illustrate the proposed extension of the Loewner framework through one simple example ($n_{\mathrm{p}} = 1$ and $N = 2$). The associated generalized observability and controllability matrices are put together as follows*

$$
\begin{aligned}
\mathscr{O} &= \begin{bmatrix} C\Phi(\mu_0) \\ C\Phi(\mu_0)A_1\Phi(\mu_1) \end{bmatrix},\\
\mathscr{R} &= \begin{bmatrix} \Phi(\lambda_0)B & \Phi(\lambda_1)A_1\Phi(\lambda_0)B \end{bmatrix}.
\end{aligned}
\tag{40}
$$

The next step is to show that we can interpret matrices:

$$\hat{E} = \mathscr{O}\mathscr{R},\ \hat{A}_0 = \mathscr{O}A_0\mathscr{R},\ \hat{A}_1 = \mathscr{O}A_1\mathscr{R},\ \hat{B} = \mathscr{O}B,\ \hat{C} = C\mathscr{R},$$

in terms of data, i.e., measurements of transfer functions. To do so, we repeat the general procedure presented in Section 5.1 for this simplified scenario, and hence write that

$$
\begin{aligned}
\hat{E} &= -\begin{bmatrix} \frac{H_0^{\mathfrak{F}}(\mu_0)-H_0^{\mathfrak{F}}(\lambda_0)}{\mu_0-\lambda_0} & \frac{H_1^{\mathfrak{F}}(\lambda_0,\mu_0)-H_1^{\mathfrak{F}}(\lambda_0,\lambda_1)}{\mu_0-\lambda_1} \\ \frac{H_1^{\mathfrak{F}}(\mu_1,\mu_0)-H_1^{\mathfrak{F}}(\lambda_0,\mu_0)}{\mu_1-\lambda_0} & \frac{H_{1,1}^{\mathfrak{F}}(\lambda_0,\mu_1,\mu_0)-H_{1,1}^{\mathfrak{F}}(\lambda_0,\lambda_1,\mu_0)}{\mu_1-\lambda_1} \end{bmatrix},\\
\hat{A}_0 &= -\begin{bmatrix} \frac{\mu_0 H_0^{\mathfrak{F}}(\mu_0)-\lambda_0 H_0^{\mathfrak{F}}(\lambda_0)}{\mu_0-\lambda_0} & \frac{\mu_0 H_1^{\mathfrak{F}}(\lambda_0,\mu_0)-\lambda_1 H_1^{\mathfrak{F}}(\lambda_0,\lambda_1)}{\mu_0-\lambda_1} \\ \frac{\mu_1 H_1^{\mathfrak{F}}(\mu_1,\mu_0)-\lambda_0 H_1^{\mathfrak{F}}(\lambda_0,\mu_0)}{\mu_1-\lambda_0} & \frac{\mu_1 H_{1,1}^{\mathfrak{F}}(\lambda_0,\mu_1,\mu_0)-\lambda_1 H_{1,1}^{\mathfrak{F}}(\lambda_0,\lambda_1,\mu_0)}{\mu_1-\lambda_1} \end{bmatrix},\\
\hat{A}_1 &= \begin{bmatrix} H_1^{\mathfrak{F}}(\lambda_0,\mu_0) & H_{1,1}^{\mathfrak{F}}(\lambda_0,\lambda_1,\mu_0) \\ H_{1,1}^{\mathfrak{F}}(\lambda_0,\mu_1,\mu_0) & H_{1,1,1}^{\mathfrak{F}}(\lambda_0,\lambda_1,\mu_1,\mu_0) \end{bmatrix},\\
\hat{B} &= \begin{bmatrix} H_0^{\mathfrak{F}}(\mu_0) \\ H_1^{\mathfrak{F}}(\mu_1,\mu_0) \end{bmatrix},\quad \hat{C} = \begin{bmatrix} H_0^{\mathfrak{F}}(\lambda_0) & H_1^{\mathfrak{F}}(\lambda_0,\lambda_1) \end{bmatrix}.
\end{aligned}
\tag{41}
$$

So, in this simple case in which $N = n_{\mathrm{p}} = 1$, it follows that $(N+1)^2 + n_{\mathrm{p}}(N+1)^2 = 8$ interpolation conditions are satisfied by the reduced model $\tilde{\Sigma}$ calculated according to (28) and (27). Below, we enumerate the transfer function values that are matched:

$$
\begin{aligned}
&H_0^{\tilde{\mathfrak{F}}}(\mu_0),\ H_0^{\tilde{\mathfrak{F}}}(\lambda_0),\ H_1^{\tilde{\mathfrak{F}}}(\mu_1,\mu_0),\ H_1^{\tilde{\mathfrak{F}}}(\lambda_0,\mu_0),\ \ H_1^{\tilde{\mathfrak{F}}}(\lambda_0,\lambda_1),\\
&H_{1,1}^{\tilde{\mathfrak{F}}}(\lambda_0,\mu_1,\mu_0),\ H_{1,1}^{\tilde{\mathfrak{F}}}(\lambda_0,\lambda_1,\mu_0),\ \ H_{1,1,1}^{\tilde{\mathfrak{F}}}(\lambda_0,\lambda_1,\mu_1,\mu_0).
\end{aligned}
$$

6 Numerical example

In this section we revisit the example presented in [30] (Section III, Example 1). Based on Assumption (1) that was imposed in Section 2 of the current paper, the B and C matrices will be considered to be constant. Additionally, we shift the original matrix A_0 from [30] so that all its poles are located into the left-half (complex) plane. Finally, choose $n_{\mathrm{p}} = 2$ (originally, $n_{\mathrm{p}} = 1$ was enforced) and assume zero initial conditions. The system matrices of the modified system are given as follows:

$$
\begin{aligned}
A_0 &= \begin{bmatrix} -1 & 1 & -1 \\ -1 & -2 & 1 \\ -1 & 1 & -3 \end{bmatrix},\ A_1 = \begin{bmatrix} 1 & -1 & -1 \\ -1 & 2 & 0 \\ -1 & 0 & 2 \end{bmatrix},\ B_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix},\\
A_2 &= \begin{bmatrix} 0 & -1 & 1 \\ 0 & 1 & 2 \\ 2 & 1 & 0 \end{bmatrix},\ C_0 = \begin{bmatrix} 1 & -1 & -1 \end{bmatrix},\ D = 0.
\end{aligned}
\tag{42}
$$

The control input is chosen as $u(t) = 0.1\cos(20t)\cdot e^{-0.1t}$, while the scheduling signals are purely oscillatory, with different main frequencies, i.e., $p_1(t) = 2.5\sin(5\pi t)$ and $p_2(t) = 1.25\sin(7\pi t)$. We apply the newly-proposed method for $N = 2$ and the following choice of left and right interpolation points (located on the imaginary axis; here $\imath = \sqrt{-1}$).

$$
\begin{cases} \mu_0 = 2\imath,\ \mu_1 = 4\imath,\ \mu_2 = 6\imath, \\ \lambda_0 = 3\imath,\ \lambda_1 = 5\imath,\ \lambda_3 = 8\imath. \end{cases}
\tag{43}
$$

It is to be noted that we construct three reduced-order models of dimension r for all values $1 \le r \le 3$, by following the procedure outlined in Section 5. The accuracy of these interpolation-based surrogate models is tested by means of time-domain simulations. We simulate the original system together with the three reduced ones on a time range of $[0,10]$s (by applying a classical first-order Euler scheme on $5\cdot 10^4$ points). The observed outputs of the original system, together with the outputs of the three reduced models are depicted in Fig. 1.

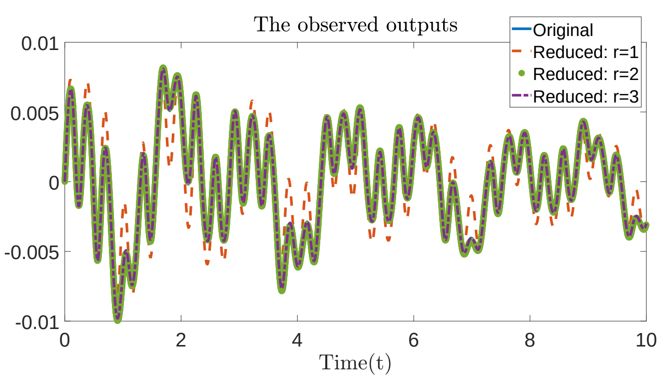

Figure 1: The observed outputs (original and reduced).

Additionally, we compute the magnitude of the relative approximation error for each reduced dimension $r \in \{1,2,3\}$ and depict the curves in Fig. 2. Clearly, the order $r = 3$ system computed by means of the new method perfectly matches the response of the original system (the approximation errors are in the range of machine precision). The other two reduced systems, of course enforce higher errors; in particular, the output of the one of order $r = 2$ follows quite accurately the original response (as illustrated in Fig. 1).

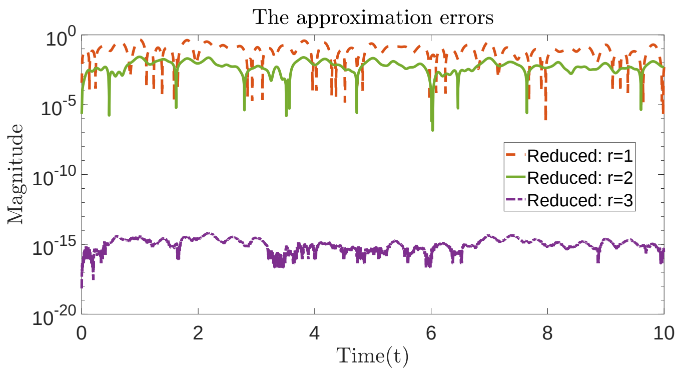

Figure 2: The relative approximation errors.

7 Conclusion

We have proposed an extension of the Loewner framework to LPV systems with an affine dependence on parameters. The proposed framework yields a model reduction procedure which is based on matching the frequency response of the original system at some particular frequencies. In order to avoid complex notations, we have restricted the attention to the single input case and to models for which the B and C matrices do not depend on the scheduling parameters. Moreover, we analyzed a particular choice of frequencies to be matched. Future research will be directed towards extending these results to general LPV systems with affine dependence on parameters. Other research directions include finding system theoretic interpretations for the proposed method, i.e., showing that for certain inputs and scheduling signals the time-domain responses of the original and reduced model coincide, possibly after filtering. Finally, we plan to test the proposed method for more complex models.

References

[1] W. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401–1425, 2000.

[2] J. Mohammadpour and C. W. Scherer, *Control of Linear Parameter Varying Systems with Applications*. Heidelberg: Springer, 2012.

[3] D. Vizer, G. Mercère, O. Prot, E. Laroche, and M. Lovera, "Linear fractional LPV model identification from local experiments: an H_∞-based optimization technique," in *In IEEE Conference on Decision and Control*, Florence, Italy, December 2013.

[4] R. Tóth, "Modeling and identification of linear parameter-varying systems," in *Lecture Notes in Control and Information Sciences, Vol. 403*. Heidelberg: Springer, 2010.

[5] R. Tóth, H. S. Abbas, and H. Werner, "On the state-space realization of LPV input-output models: Practical approaches," *IEEE Trans. Contr. Syst. Technol.*, vol. 20, pp. 139–153, Jan. 2012.

[6] B. Bamieh and L. Giarré, "Identification of linear parameter varying models," *International Journal of Robust and Nonlinear Control*, vol. 12, pp. 841–853, 2002.

[7] J. W. van Wingerden and M. Verhaegen, "Subspace identification of bilinear and LPV systems for open- and closed-loop data," *Automatica*, vol. 45, no. 2, pp. 372–381, 2009.

[8] P. L. dos Santos, J. A. Ramos, and J. L. M. de Carvalho, "Identification of LPV systems using successive approximations," in *Proc. of 47th IEEE Conference on Decision and Control*, 2008, pp. 4509–4515.

[9] M. Sznaier and C. Mazzaro, "An LMI approach to the identification and (in)validation of LPV systems," in *Perspectives in robust control. Lecture Notes in Control and Information Sciences*, S. Moheimani, Ed. London: Springer, 2001, vol. 268, pp. 327–346.

[10] V. Verdult and M. Verhaegen, "Subspace identification of multivariable linear parameter-varying systems," *Automatica*, vol. 38, no. 5, pp. 805–814, 2002.

[11] F. Blanchini, D. Casagrande, S. Miani, and U. Viaro, "Stable LPV realization of parametric transfer functions and its application to gain-scheduling control design," *IEEE Transactions on Automatic Control*, vol. 55, no. 10, pp. 2271–2281, 2010.

[12] A. C. Antoulas, *Approximation of large-scale dynamical systems*, ser. Advances in Design and Control. SIAM, 2005.

[13] P. Benner, S. Gugercin, and K. Willcox, "A survey of projection-based model reduction methods for parametric dynamical systems," *SIAM Review*, vol. 57, no. 4, pp. 483–531, 2015.

[14] A. C. Antoulas, C. Beattie, and S. Güğercin, *Interpolatory methods for model reduction*, ser. Computational Science and Engineering 21. SIAM, Philadelphia, 2020.

[15] M. Farhood, C. Beck, and G. Dullerud, "On the model reduction of nonstationary LPV systems," in *Proc. of the American Control Conference (ACC)*, Denver, CO, USA, Jun. 2003, pp. 3869 – 3874.

[16] S. D. Hillerin, G. Scorletti, and V. Fromion, "Reduced-complexity controllers for LPV systems: Towards incremental synthesis," in *Proc. of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, Orlando, FL, USA, Dec. 2011, pp. 3404 – 3409.

[17] F. Adegas, I. Sonderby, M. Hansen, and J. Stoustrup, "Reduced-order lpv model of flexible wind turbines from high fidelity aeroelastic codes," in *Proc. of the IEEE International Conference on Control Applications (CCA)*, Hyderabad, Aug. 2013, pp. 424 – 429.

[18] G. Wood, P. Goddard, and K. Glover, "Approximation of linear parameter-varying systems," in *Proc. of the 35th IEEE Conference on Decision and Control*, Kobe, Dec. 1996, pp. 406 – 411.

[19] Widowati, R. Bambang, R. Saragih, and S. Nababan, "Model reduction for unstable LPV systems based on coprime factorizations and singular perturbation," in *Proc. of the 5th Asian Control Conference*, Melbourne, Jul. 2004, pp. 963 – 970.

[20] M. Bastug, M. Petreczky, R. Tóth, R. Wisniewski, J. Leth, and D. Efimov, "Moment matching based model reduction for lpv state-space models," in *Decision and Control (CDC), 2015 IEEE 54rd Annual Conference on*, 2015.

[21] P. Benner, X. Cao, and W. Schilders, "A bilinear H_2 model order reduction approach to linear parameter-varying systems," *Adv Comput Math*, vol. 45, pp. 2241–2271, 2019.

[22] S. Z. Rizvi, J. Mohammadpour, R. Tóth, and N. Meskin, "A kernel-based PCA approach to model reduction of linear parameter-varying systems," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 5, pp. 1883–1891, 2016.

[23] T. Luspay, T. Péni, I. Gözse, Z. Szabó, and B. Vanek, "Model reduction for LPV systems based on approximate modal decomposition," *International Journal for Numerical Methods in Engineering*, vol. 113, no. 6, pp. 891–909, 2018.

[24] S. Schouten, D. Lou, and S. Weiland, "Model reduction for linear parameter-varying systems through parameter projection," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 7800–7805.

[25] A. Mayo and A. Antoulas, "A framework for the solution of the generalized realization problem," *Linear Algebra and Its Applications*, vol. 425, no. 2-3, pp. 634–662, 2007.

[26] I. V. Gosea, M. Petreczky, and A. C. Antoulas, "Data-driven model order reduction of linear switched systems in the Loewner framework," *SIAM Journal on Scientific Computing*, vol. 40, no. 2.

[27] A. C. Antoulas, I. V. Gosea, and A. C. Ionita, "Model reduction of bilinear systems in the Loewner framework," *SIAM Journal on Scientific Computing*, vol. 38(5), pp. B889–B916, 2016.

[28] W. J. Rugh, *Linear System theory*. Prentice-Hall, 1996.

[29] M. Siraj, R. Toth, and S. Weiland, "Joint order and dependency reduction for LPV state-space models," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 2012, pp. 6291–6296.

[30] M. Petreczky, G. Mercère, and R. Tóth, "Affine LPV systems : realization theory , input-output equations and relationship with linear switched systems," *IEEE Transactions on Automatic Control*, vol. 62, pp. 4667–4674, 2017.

[31] R. Webster, *Convexity*. Oxford, 1994.

[32] A. Isidori, *Nonlinear Control Systems*. Springer Verlag, 1989.

[33] A. C. Antoulas, S. Lefteriu, and A. C. Ionita, "A tutorial introduction to the Loewner framework for model reduction," in *Model Reduction and Approximation*. SIAM, 2017, ch. 8, pp. 335–376.