Preprints, Joint 9th IFAC Symposium on Robust Control Design and
2nd IFAC Workshop on Linear Parameter Varying Systems
Florianopolis, Brazil, September 3-5, 2018

Transformations for linear parameter varying systems ⋆

Z. Szabó * **J. Bokor** **

* *Institute for Computer Science and Control, Hungarian Academy of Sciences, Budapest, Kende u. 13-17, Hungary, (Tel: +36-1-279-6171; e-mail: szabo.zoltan@sztaki.mta.hu).*
** *Institute for Computer Science and Control, Hungarian Academy of Sciences, Budapest, Kende u. 13-17, Hungary*

Abstract: The LPV modelling paradigm grew up from the desire of having a gain scheduling method with guaranteed stability and performance bound by using as much as possible from the LTI design techniques. In the last decades the framework has proven its applicability in the field of robust control design. Some basic modelling issues, however, such as system equivalence, state transformation, loop transformation, does not gain much attention. The main goal of the paper is to provide an initialization of the novices in LPV modelling in order to eliminate the possible pitfalls that still often occur in the related literature. On the other side, we would like to point out some potential research topics that might also be interesting for a much larger audience.

Keywords: LPV system, state transformation, loop transformation.

1. INTRODUCTION AND MOTIVATION

Rooted in the idea of gain scheduling, linear parameter varying (LPV) modelling has proven to be an efficient approach in many areas of control and filtering in treating nonlinear problems in the past decades. A broad class of nonlinear system models can be converted into a quasi-linear form, obtaining the system:

$$\dot{x}(t) = A(\rho)x(t) + B(\rho)u(t), \quad x(0) = 0, \tag{1}$$

$$y(t) = C(\rho)x(t) + D(\rho)u(t), \tag{2}$$

where $x \in \mathcal{X} \subset \mathbb{R}^n$ is the state, $u \in \mathbb{R}^m$ and $y \in \mathbb{R}^p$ are the input and output functions, respectively, while $\rho \in \mathbf{\Omega}$ is the vector of scheduling functions, which are determined by the measured variables. This means that their values are known in operational time by measurement. The approach is particularly appealing when a natively nonlinear problem, embedded in the LPV framework, can be solved by using traditional linear techniques.

Depending the actual model, the collection of the allowed scheduling variable might vary from a subset $\mathbf{\Omega}$ of the measurable functions (when we actually have a switched system) to a subset of constant functions (when we actually have a class of LTI systems: more precisely an LTI system with some uncertain parameters). Concerning the topic of this paper the later class is completely uninteresting. Moreover, the worst mistake that one could do when dealing with LPV systems is to confuse it with a collection of LTI systems.

In order to decrease the conservativeness of the design, often the elements of $\mathbf{\Omega}$ are also supposed to be sufficiently smooth, taking values from a compact set $\mathcal{P}$. Usually smooth means that the scheduling parameter is of class C^1, i.e., it has a continuous derivative. It is a standard assumption that $\mathcal{P}$ is of box type, i.e., each parameter ρ_i ranges between its known extremal values $\rho_i(t) \in [\underline{\rho}_i, \overline{\rho}_i]$. While the derivatives of the scheduling variables usually are not measured, in control design problems they are supposed to be bounded, i.e., $\rho^{(i)}(t) \in \mathcal{P}_i \subseteq \mathbb{R}^{n_\rho}$. Typically $i = 1$. We will denote by $\mathbf{\Omega}_\infty$ the case when the scheduling variables are measurable functions taking values from $\mathcal{P}$, while $\mathbf{\Omega}_1$ stands for the case when the scheduling variables are smooth, their values being constrained by the condition $(\rho, \dot{\rho}) \in \mathcal{P} \times \mathcal{P}_1$, respectively.

While during the last decades the framework has proven its applicability in the field of robust control design, some basic modelling issues, such as system equivalence, state and more generally, loop, transformation, does not gain much attention. Constant state transformations are intimately related to the concept of invariant subspace known from the geometric theory of LTI systems and it were extended to LPV dynamics by introducing the notion of parameter-varying invariant subspace, see Balas et al. (2003). In introducing the various parameter-varying invariant subspaces an important goal was to set notions that lead to computationally tractable algorithms for the case when the parameter dependency of the system matrices is affine. These invariant subspaces play the same role in the solution of the fundamental problems, such as disturbance decoupling, unknown input observer design, fault detection, as their counterparts in the time invariant context, see Szabó et al. (2003); Bokor and Balas (2004).

State transformations provide a tool to define or, which is more important from a practical point of view, to test the equivalence of the representations of type (1)-(2). In

⋆ This work has been supported by the GINOP-2.3.2-15-2016-00002 grant of the Ministry of National Economy of Hungary and by the European Commission through the H2020 project EPIC under grant No. 739592.

the context of the LPV framework it is desirable to also apply parameter varying transformations, e.g., for model reduction tasks, Luspay et al. (2018). In contrast to the LTI case, the issue has shown to be highly nontrivial, see, e.g., Alkhoury et al. (2017).

Traditionally the LPV framework is formulated in terms of a state space representations. There is a possibility, however, to develop a sound input-output (I/O) description related to this model class. For a motivating example see the LPV null space generator methods, Szabó et al. (2015), and its applications to fault detection and reconfiguration, Péni et al. (2017). In this paper we present a general stability preserving loop transformation result which, among others, describes the parameter varying state space transformation, Youla parametrization, etc.

The main goal of the paper is to provide an initialization of the novices in LPV modelling in order to eliminate the possible pitfalls that still often occur in the related literature. We focus on two transformation techniques: state and loop transformation, respectively, and we visit a series of issues related to these transformations specific for the LPV model class. As a result of this challenging travel we would like to point out and to formulate some potential research topics related to the LPV modelling framework that might also be interested for a much larger audience.

Section 2 shows how an LPV system can be considered as an I/O operator. Through elementary examples we highlight the main difference between the LTI view and the LPV framework. Section 3 points out that the parameter varying (time varying) state transformations inherently violate the causality requirement imposed in the definition of the model class. The main question that we can formulate at this point is how to eliminate the derivatives of the scheduling variable, if it is possible, from the state matrices.

In the second part of the paper Section 4 revisits some stability issues related to LPV and recalls the fundamental results concerning the double coprime factorization and related Youla parametrization in the LPV context. In Section 5 we formualate the general loop transformation result and we emphasise its role in the development of robust control results.

2. LPV VS. I/O FRAMEWORK

Viewed at a fixed parameter trajectory ρ, the linear system that obeys to (1)-(2) can be cast as a linear time varying (LTV) system. Thus, it is convenient to consider the LPV system as a collection of time varying systems:

$$\left\{ \Sigma(\rho) \sim \left[\begin{array}{c|c} A(\rho) & B(\rho) \\ \hline C(\rho) & D(\rho) \end{array}\right] \mid \rho \in \mathbf{\Omega} \right\}. \tag{3}$$

While this embedding of the LPV plant as a class of LTV systems bears significant advantages there are some issues that evades the LTV framework, which will be highlighted next in the paper. Nevertheless, by a slight abuse of the notation, in what follows we identify and denote the LPV system as $(\Sigma(\rho), \mathbf{\Omega})$. If it is clear from the context, we will drop $\mathbf{\Omega}$ from the notation. We emphasise, however, that the parameter set – as a collection of different parameter trajectories – is an essential part in the definition of the LPV system.

LTI systems are often represented through their transfer functions. While transfer functions are tight to frequency domain, nothing prevents us to identify them with time domain input-output (I/O) operators that stands for the same LTI system. In the time varying context we do not have a sound frequency domain description and transfer functions. Nevertheless, the idea can be extended to LTV systems and thus to a collection of LTV systems parametrized through some $\rho \in \mathbf{\Omega}$, hence, to an LPV system. In this sense we can talk on $\Sigma(\rho)$ as an I/O map, regardless to the possible/actual state space representation.

Thus, the well known algebraic operations as $\Gamma(\rho_1)+\Sigma(\rho_2)$ or $\Gamma(\rho_1)\Sigma(\rho_2)$, make sense among the corresponding LPV systems provided that the input(output) dimensions are compatible.

Classical LTI realization theory that links transfer functions to state space descriptions does not have an LPV counterpart. However, the results of the classical LTV realization theory, which links the zero-initial state input-output representations of the form

$$y(t) = \int_0^t K(t,\tau)u(\tau)\mathrm{d}\tau$$

to the state matrices $(A(t), B(t), C(t))$, see, e.g., Kalman (1963); Silverman (1966); Isidori and Ruberti (1976); Kamen (1979); Sontag (1979); Dewilde and van der Veen (1998) just to mention a few of the dozens of relevant accounts, are applicable, see, e.g., Tóth (2010); Tóth et al. (2012); Petreczky et al. (2017). At this point we would like to recall only one significant element related to the topic, namely the equivalence of different representations.

For LTV systems Kalman (1963) sets the fundamental concept: a state transformation $\xi = T(t)x$ with nonsingular $T(t)$ on the time axis defines an equivalent system (algebraic equivalence), while if both $T(t)$ and $T^{-1}(t)$ is bounded we have a so called topological equivalence that preserves stability. If the system matrices are sufficiently smooth, Silverman and Meadows (1969) provides additional details. Considering the matrices

$$P_{i+1}(t) = -A(t)P_i(t) + \frac{\mathsf{d}}{\mathsf{d}t}P_i(t), \quad P_0(t) = B(t),$$
$$S_{i+1}(t) = S_i(t)A(t) + \frac{\mathsf{d}}{\mathsf{d}t}S_i(t), \quad S_0(t) = C(t),$$

and defining the corresponding $Q_k(t) = [P_0(t) \ \cdots \ P_{k-1}(t)]$ controllability and $R_k^T(t) = \left[S_0^T(t) \ \cdots \ S_{k-1}^T(t)\right]$ observability matrices, respectively, the constant rank system representation is completely controllable (observable) if there exists integers α (β) such that if $\mathrm{rank}Q_k(t) = n$ ($\mathrm{rank}R_k(t) = n$) for all t and $k \geq \alpha$ ($k \geq \beta$), where n is the dimension of the state. Two controllable constant rank system representations (A, B, C) and $(\bar{A}, \bar{B}, \bar{C})$ of order n are algebraically equivalent if and only if $\bar{P}_k(t) = T(t)P_k(t)$ and $\bar{C}(t) = C(t)T^{-1}(t)$, i.e, $T(t) = \bar{Q}_\gamma(t)Q_\gamma^\dagger(t)$ for all t, and $\gamma = \max\{\bar{\alpha}, \alpha\}$. By duality we have an analogous statement.

Note, that while it is hard to test it in practice, in the context of LPV systems it is desirable to have a constant rank representation for every ρ. E.g., considering the restriction of $\mathbf{\Omega}$ to constant functions and the minimality of the resulting LTI representations it is desirable to have the same state space dimension.

We conclude this section by arguing against a bad habit to mix transfer function notations with time domain scheduling variables of type $G(s,\rho(t))$. The next example reveals that there is no sound interpretation of such formulas even if s is interpreted as time differentiation: consider the system

$$\dot{y}(t) = -\alpha y(t) + \rho(t)u(t). \tag{4}$$

Associated to this system we often encounter the notation $\frac{\rho(t)}{s+\alpha}$ whose interpretation is ambiguous.

The key point here is that the differentiation operator does not commute with the multiplication operator defined by time varying functions. In particular

$$\rho(t)\cdot\frac{1}{s+\alpha} \neq \frac{1}{s+\alpha}\cdot\rho(t).$$

Thus, the systems

$$\dot{x} = -\alpha x + \rho(t)u(t), \quad y(t) = x(t) \tag{5}$$

and

$$\dot{x} = -\alpha x + u(t), \quad y(t) = \rho(t)x(t) \tag{6}$$

are different. This small example also reveals the fact that in contrast to the misbelieve often encountered in some papers, by merely defining some LTI systems on a given parameter grid does not define an LPV system. Further examples can be found in Blanchini et al. (2010). To define an LPV system a rule is also necessary, that uniquely provides the system matrices in every frozen parameter point. This rule is often a linear interpolation.

We emphasize, that there is a fundamental difference between the interpretation sketched above and the case when an operator, which for convenience might be denoted by $1/s$, enters in a linear fractional transform (LFT), e.g.,

$$D(\rho) + C(\rho)(1/s)[I - A(\rho)(1/s)]^{-1}B(\rho). \tag{7}$$

The latter stands for the following set of constraints:

$$\begin{aligned} \eta &= A(\rho)\xi + B(\rho)u \\ y &= C(\rho)\xi + D(\rho)u, \\ \xi &= (1/s)\eta, \quad \text{i.e.,} \quad \xi(t) = \int_0^t \eta(\tau)\mathrm{d}\tau, \end{aligned}$$

provided that the loop make sense (is well-posed). Note that in contrast to (7) the notation $G(s,\rho(t))$ can not imply a priori any particular realization in the LPV context, i.e., any particular LPV system.

3. STATE TRANSFORMATIONS

Having an LPV system it is natural to consider parameter varying state transformations, i.e., $\tilde{x} = T(\rho)x$ for $\rho \in \mathbf{\Omega}$ that leads to

$$\tilde{\Sigma}(\rho) \sim \left[\begin{array}{c|c} \tilde{A}(\rho,\dot{\rho}) & \tilde{B}(\rho) \\ \hline \tilde{C}(\rho) & \tilde{D}(\rho) \end{array}\right] = \tag{8}$$

$$= \left[\begin{array}{c|c} T(\rho)A(\rho)T^{-1}(\rho) + \dot{T}(\rho)T^{-1}(\rho) & T(\rho)B(\rho) \\ \hline C(\rho)T^{-1}(\rho) & D(\rho) \end{array}\right],$$

provided that the scheduling variables are smooth.

We arrive here to some problematic points which makes a clear difference between LTV and LPV systems. If ρ is not smooth, e.g., the LPV system is in class $\mathbf{\Omega}_\infty$, then state transformations might send the system description to a different class, namely to the class of impulsive systems.

As we have already seen, if the system is sufficiently smooth, the system equivalence is exhausted by transformations that might depend up to the $(n-1)^{\text{th}}$ derivative of the scheduling variable. Even ρ is supposed to be smooth, such a state transformation will send our description outside the LPV framework, in general.

Actually there are two problems here. The first problem is more apparent: even if we allow reparametrization (inflation of the parameter space) the type of the problem might change. Starting from an $\mathbf{\Omega}_1$ type system we might obtain an $\tilde{\mathbf{\Omega}}_\infty$ type of system.

The second problem is more subtle: derivation is not a causal operation, thus we violate our assumption on the availability of the information. To make this point more clear, consider the same transformation in discrete time:

$$A(\rho_k) \mapsto T(\rho_{k+1})A(\rho_k)T^{-1}(\rho_k), \quad B(\rho_k) \mapsto T(\rho_{k+1})B(\rho_k)$$

As an illustration we slightly modify an example from Tóth et al. (2007): consider the input–output map defined by

$$y(k) = -\rho(k-1)y(k-1) - \rho(k-1)y(k-2) + \rho(k-1)u(k-1)$$

(the original example have used $\rho(k)$). Considering $x(k) = \begin{bmatrix} y(k-1) \\ y(k) \end{bmatrix}$ as state, it is immediate that

$$\Sigma(\rho) \sim \left[\begin{array}{cc|c} 0 & 1 & 0 \\ -\rho(k) & -\rho(k) & \rho(k) \\ \hline 0 & 1 & 0 \end{array}\right]$$

is a realization. It is less trivial that the reachability (observability) canonical realizations for the same system are provided by

$$\Sigma^c(\rho) \sim \left[\begin{array}{cc|c} 0 & -\rho(k-2) & 1 \\ 1 & -\rho(k-2) & 0 \\ \hline \rho(k-1) & -\rho(k-1)\rho(k-2) & 0 \end{array}\right]$$

and

$$\Sigma^o(\rho) \sim \left[\begin{array}{cc|c} 0 & 1 & \rho(k) \\ -\rho(k+1) & -\rho(k+1) & -\rho(k+1)\rho(k) \\ \hline 1 & 0 & 0 \end{array}\right],$$

respectively.

Concerning continuous time systems, revisit (9) and (10) applying the state transform defined by $\xi = \rho x$ to obtain:

$$\dot{\xi} = (-\alpha + \dot{\rho}/\rho)\xi + \rho^2(t)u(t), \quad y(t) = (1/\rho)\xi(t) \tag{9}$$

and

$$\dot{\xi} = (-\alpha + \dot{\rho}/\rho)\xi + \rho u(t), \quad y(t) = \xi(t), \tag{10}$$

respectively.

These examples clearly reveal an aspect related to the LPV modelling that remains in shadow up till now: identification approaches ignore causality, as an important property of the model class, i.e., the possibility to implement it. In control design applications this is definitely a requirement: $\dot{\rho}$ $(\rho(k+1))$ is not available. One could argue that a remedy were a reformulation of the model class by requiring the availability of the necessary signals. In practice this would mean, for example, the use of a suitably filtered scheduling variable ρ_f where $\dot{\rho}_f$ (or even ρ_f^k) would be also available. The introduced delay makes such an artefact useless for control purposes, in general.

Moreover, identification approaches often assume that the LPV system is given in a certain structure (e.g., ARMA, controllability like form). In contrast to the LTI case it is not clear which class of LPV systems can be modelled in that way.

In contrast to the LTI case, to test equivalence of two different LPV representations is highly nontrivial. From both theoretical and practical point of views this is a big deficiency of the LPV modelling paradigm which raises the quest for further research in this direction. The practical question is how to eliminate, if it is possible, by an application of a suitable transformation the derivatives of the parameters from the state matrices.

To conclude this section we would like to reveal some problems related to the applicability of the gridding approach in this context. It was already emphasised that the knowledge of the LTI systems on a grid does not define an LPV system regardless to the resolution of the grid. Adding an interpolation method, e.g., linear, however, leads to an LPV system. One might think that the same idea can be used for the transformation, too. Unfortunately, this is not true, in general: e.g., if $\Sigma(\rho)$ and $\hat{\Sigma}(\rho)$ represent the same LPV system, defined on a grid through linear interpolation, the state transformation relating them cannot be (piecewise)linear on the same grid. For a smooth system, see, e.g., the formula $T(t) = \bar{Q}_\gamma(t)Q_\gamma^\dagger(t)$ that make this impossible.

4. STABILITY, STABILIZABILITY

Closed loop stability and as a related issue, parametrization of the controllers that renders a given plant stable, is a central topic in classical control theory. In the context of stability, causality plays a definite role: systems are stable if they define a bounded and causal map. In the standard linear model signals are elements of some normed linear spaces, the system is identified with an operator that acts between signals, while boundedness of the system is regarded as boundedness in the induced operator norm. For more details on nest algebras, causality and time varying systems, see, e.g., Feintuch (1998).

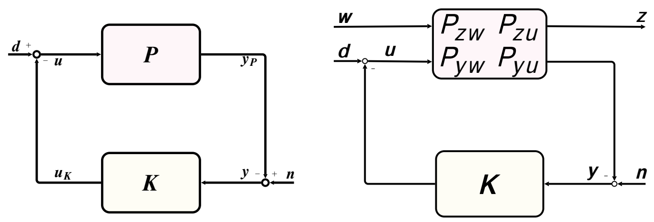
Stability of an LPV system can be defined in a straightforward manner: the LPV system is stable if for each $\rho \in \mathbf{\Omega}$ the LTV system $\Sigma(\rho)$ is stable. Observe that stability of LPV systems are tight to the parameter set. The two cases encountered in practice are $\mathbf{\Omega}_\infty$ and $\mathbf{\Omega}_1$, respectively. We prefer to term the first case as strong stability (as a hint for switched systems) and as parameter varying stability the second.

4.1 Youla parametrization

The feedback connection depicted on Figure 1(a), i.e., the pair (P, K), is called stable if for every w there is a unique p and k such that $w = p + k$ (causal invertibility) and if the map $w \to z$ is a bounded causal map, where

$$w = \begin{pmatrix} d \\ n \end{pmatrix}, p = \begin{pmatrix} u \\ y_P \end{pmatrix}, k = \begin{pmatrix} u_K \\ y \end{pmatrix}, z = \begin{pmatrix} u \\ y \end{pmatrix}.$$

Accordingly the pair (P, K) is called stable if and only if the inverse



(a) Basic loop: stability (b) LFT: performance

Fig. 1. Closed loop: performance and internal stability

$$\begin{pmatrix} I & K \\ P & I \end{pmatrix}^{-1} = \begin{pmatrix} S_u & S_c \\ S_p & S_y \end{pmatrix} = \\ = \begin{pmatrix} (I-KP)^{-1} & -K(I-PK)^{-1} \\ -P(I-KP)^{-1} & (I-PK)^{-1} \end{pmatrix} \quad (11)$$

exists and is stable, i.e., all the block elements are stable.

It follows that if a plant can be stabilized by feedback then it has a stable factorization $P = S_p S_u^{-1}$. Ii is usually assumed that among the stable factorizations there exists a special one, called double coprime factorization, i.e., $P = NM^{-1} = \tilde{M}^{-1}\tilde{N}$ and there are causal bounded systems $U, V, \tilde{U}$ and $\tilde{V}$ such that

$$\begin{pmatrix} \tilde{V} & -\tilde{U} \\ -\tilde{N} & \tilde{M} \end{pmatrix} \begin{pmatrix} M & U \\ N & V \end{pmatrix} = \tilde{\Sigma}_P \Sigma_P = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}, \quad (12)$$

an assumption which is often made when setting the stabilization problem, Vidyasagar (1985); Feintuch (1998). Recall that $\begin{pmatrix} M \\ N \end{pmatrix}$ and $\begin{pmatrix} U \\ V \end{pmatrix}$ are determined only up to stably invertible factors (invertible with stable inverse) T and T'. The existence of a double coprime factorization implies feedback stabilizability. In most of the usual model classes actually there is an equivalence.

Given a double coprime factorization the set of the stabilizing controllers is provided through the well-known Youla parametrization:

$$\mathcal{K}_{stab} = \{K = \mathfrak{M}_{\Sigma_P}(Q) \mid Q \in \mathbb{Q}, (V + NQ)^{-1} \text{ exists}\},$$

where $\mathbb{Q} = \{Q \,|\, Q \text{ stable }\}$ and

$$\mathfrak{M}_{\Sigma_P}(Q) = (U + MQ)(V + NQ)^{-1}.$$

Note that $Q = \mathfrak{M}_{\tilde{\Sigma}_P}(K)$ and thus $Q = 0_K$ corresponds to $K_0 = UV^{-1}$. Since the dimensions of the controller and plant are different, it is convenient to distinguish the zero controller and zero plant by an index, i.e., 0_K and 0_P, respectively.

It is obvious that the entire scheme remains valid for the LPV framework, too. See the Appendix for an illustration of the relevant calculus. The only constraint is to respect the stability concept set by the given LPV model, i.e., by the parameter set. In practice one can use either the strong stability or the parameter varying stability, when selecting the elements of the parametrization.

Closely related to stability is the concept of stabilizability, i.e., the ability to obtain a stabilizing controller K. For practical reasons this concept is traditionally closely related to a state space representation of the linear system and it boils down to finding a stabilizing parameter varying state feedback gain.

Recall that, analogous to the LTI case, having a stabilizing state feedback gain F and a stabilizing output injection gain L one has

$$\Sigma_P(\rho) \sim \left[\begin{array}{c|cc} A(\rho)+B(\rho)F(\rho) & B(\rho) & -L(\rho) \\ \hline F(\rho) & I & 0 \\ C(\rho) & 0 & I \end{array}\right] \tag{13}$$

At this point, due to the embedding of the LPV systems into the class of LTV systems, we can talk on asymptotically, exponentially or uniform exponentially stabilizable systems, see, e.g., Anderson et al. (2013).

Some authors prefer to qualify stability of LPV systems, and hence the corresponding Youla parametrization, according to our ability to provide for them a (quadratic) Lyapunov function, see, e.g., Xie and Eisaka (2004). Thus, if we have a stability guarantee for some $\mathbf{\Omega}_\infty$ proved by a constant Lyapunov matrix then we have quadratic stability. For $\mathbf{\Omega}_1$ a parameter varying Lyapunov matrix is associated, and the corresponding stability is termed as parameter varying quadratic stability. Note, that these stability tests provide only sufficient conditions.

Finally, note, that the entire construction has a considerable freedom in the choice of the given elements, like Σ_P and Q, which makes possible to embed a given system in different frameworks. The standard example is to let the parameter Q to be a stable LPV system obtaining an LPV controller. But nothing prevents us to set also M and N to be LPV systems even the original system was an LTI one. To achieve this, it is sufficient to consider parameter varying gains in (13). Thus, we obtain an example when $N(\rho)M^{-1}(\rho)$ leads to an LTI system, i.e., there is a parameter cancellation effect.

If we get rid of the actual context of coprime factorizations it is possible to formulate another research question: given an LPV system $\Sigma(\rho)$ under which condition is it possible to impose "parameter cancellation", i.e., to eliminate some (or the entire) parameter dependence by a suitable filtering from the resulting LPV system $\Sigma(\rho)\Gamma(\rho)$.

5. LOOP TRANSFORMATIONS

Besides the application of a suitable factorization, the technique that leads to the Youla parametrization is closely related to the application of a loop transformation, that relates the original configuration with an other one, with possible simple structure, in such a way that the stability properties are kept intact.

In robust control problems often it is convenient to perform loop-transformations, i.e., to consider maps between controller sets that are defined by Möbius transformations. These loop-transformations are also intimately related to different factorizations, that simplify the structure of the problem. Since a (robust) performance problem can be handled in the robust stability framework, these transformations are also relevant in a much wider context.

In what follows we present a result that reveals under what conditions the internal stability of the loop is preserved by performing a loop transformation defined by a Möbius transform. For convenience, the controller K is transformed; the other case (Δ in a $\Delta - P - K$ structure) can be obtained by using straightforward manipulations.

This question has already got a partial answer in Ball et al. (1991) based on the scattering approach through the Potapov-Ginsburg transformation $\hat{P}$ of the generalized plant P. However, that method should assume a left or right invertibility of P and does not provide an explicit formula for the transformed configuration.

To fix the notations, the lower and an upper LFT is defined as

$$\mathfrak{F}_l(P_g, K) = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw}$$

and

$$\mathfrak{F}_u(P_g, \Delta) = P_{yu} + P_{yw}\Delta(I - P_{zw}\Delta)^{-1}P_{zu}.$$

In the generalized plant paradigm the loop should be stable and the resulting system should satisfy some norm constraints. In general, stability of the LFT loop means that the causal map that relates the signals (z, u, y) to (w, d, n) is invertible and the inverse map is stable, see Figure 1(b).

Let us consider the linear map $\mathcal{T} : \begin{pmatrix} z \\ y \end{pmatrix} \mapsto \begin{pmatrix} w \\ u \end{pmatrix}$, and its inverse (if exists) described by the operator matrices

$$\mathcal{T} = \begin{pmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{pmatrix}, \quad \text{and} \quad \mathcal{T}^{-1} = \begin{pmatrix} \mathcal{E} & \mathcal{F} \\ \mathcal{G} & \mathcal{H} \end{pmatrix}, \tag{14}$$

respectively. We will use this notation throughout the rest of the paper.

Möbius transformations, which are usually defined as

$$Z' = \mathfrak{M}_{\mathcal{T}}(Z) = (\mathcal{C} + \mathcal{D}Z)(\mathcal{A} + \mathcal{B}Z)^{-1},$$

relate two graph subspaces through the invertible linear operator $\mathcal{T}$ on the domain

$$\mathrm{dom}\mathfrak{M}_{\mathcal{T}} = \{(\mathcal{A} + \mathcal{B}Z)^{-1} \text{ exists } \}.$$

Thus they inherit the group structure of the linear operators, i.e.,

$$\mathfrak{M}_{\mathcal{P}} \circ \mathfrak{M}_{\mathcal{Q}}(Z) = \mathfrak{M}_{\mathcal{PQ}}(Z). \tag{15}$$

provided that the corresponding expressions exist.

Analogously, one can introduce a Möbius transformation that relates inverse graph subspaces according to

$$Z'' = \mathfrak{M}_{\bar{\mathcal{T}}}(Z) = (\mathcal{A}Z + \mathcal{B})(\mathcal{C}Z + \mathcal{D})^{-1}.$$

Due to their role in the control oriented context, we associate $\mathfrak{M}_{\mathcal{T}}(P) = \mathfrak{P}_{\mathcal{T}}(P)$ with a plant transform and $\mathfrak{M}_{\bar{\mathcal{T}}}(K) = \mathfrak{K}_{\mathcal{T}}(K)$ with a controller transform, respectively.

The following result, see, Szabó et al. (2017), provides an explicit loop-transformation formula:

Theorem 1. Let us consider the transformation of the standard LFT control loop from Figure 1(b) defined by an unimodular $\mathcal{T}$ which sends K to $\hat{K} = \mathfrak{K}_{\mathcal{T}}(K)$ and we also assume that $P_{yu} \in \mathrm{dom}\mathfrak{P}_{\mathcal{T}}$. Then we have

$$\mathfrak{F}_l(P_g, K) = \mathfrak{F}_l(\hat{P}_g, \hat{K}), \tag{16}$$

where $\hat{P}_g = \begin{pmatrix} \hat{P}_{zw} & \hat{P}_{zu} \\ \hat{P}_{yw} & \hat{P}_{yu} \end{pmatrix} =$

$$\begin{pmatrix} P_{zw} - P_{zu}(\mathcal{A} + \mathcal{B}P_{yu})^{-1}\mathcal{B}P_{yw} & P_{zu}(\mathcal{A} + \mathcal{B}P_{yu})^{-1} \\ (P_{yu}\mathcal{F} - \mathcal{H})^{-1}P_{yw} & (\mathcal{C} + \mathcal{D}P_{yu})(\mathcal{A} + \mathcal{B}P_{yu})^{-1} \end{pmatrix}. \tag{17}$$

Moreover the (internal) stability of the corresponding LFT loops are equivalent.

5.1 State transform vs. loop transform

It might be surprising at the first glance that the time varying state transformation formula fit into this framework: consider $\mathfrak{F}_l(P_g, 1/s)$ with

$$P_g = \begin{pmatrix} D(\rho) & C(\rho) \\ B(\rho) & A(\rho) \end{pmatrix}$$

and apply the loop transform

$$\mathcal{T}(\rho) = \begin{pmatrix} T(\rho) & 0 \\ \dot{T}(\rho) & T(\rho) \end{pmatrix}, \quad \mathcal{T}^{-1}(\rho) = \begin{pmatrix} T^{-1}(\rho) & 0 \\ T^{-1}\dot{T}T^{-1}(\rho) & T^{-1}(\rho) \end{pmatrix}.$$

Observe that (17) gives

$$\hat{P}_g = \begin{pmatrix} D(\rho) & C(\rho)T^{-1}(\rho) \\ T(\rho)B(\rho) & T(\rho)A(\rho)T^{-1}(\rho) + \dot{T}(\rho)T^{-1}(\rho) \end{pmatrix},$$

as expected. The nontrivial part is that

$$1/s = \mathfrak{K}_{\mathcal{T}}(1/s) = T(1/s)[T + \dot{T}(1/s)]^{-1},$$

which is an easy consequence of the identity

$$\frac{\mathrm{d}}{\mathrm{d}t}(Tw) = T\dot{w} + \dot{T}w,$$

i.e.,

$$(T1/s)(\dot{w}) = (1/s)[T + \dot{T}(1/s)](\dot{w}).$$

Thus

$$\mathfrak{F}_l(P_g, 1/s) = \mathfrak{F}_l(\hat{P}_g, 1/s).$$

5.2 Loop transform and Youla parametrization

The classical Youla result for LFTs also fits into this framework: one of the key observations is that the LFT loop is stable for a K if and only if the pair $(P_g, \mathrm{diag}\{0, K\})$ is stable. If the loop is stabilizable, by fixing a particular stabilizing K_0 we have a double coprime factorization induced by the stable pair (P_{yu}, K_0) (inner loop): $K_0 = uv^{-1} = \tilde{v}^{-1}\tilde{u}$ and $P_{yu} = nm^{-1} = \tilde{m}^{-1}\tilde{n}$. Considering the unimodular matrix $T = \begin{pmatrix} m & u \\ n & v \end{pmatrix}$ it is immediate that $\hat{P}_{yu} = 0$ and $\hat{K}_0 = 0$. Moreover, it is immediate that the pair $(0, q)$ is stable if and only if q is stable. Thus, applying Theorem 1 we obtain all the stabilizing controllers of P_{yu} as $\mathfrak{K}_{T^{-1}}(q)$, i.e., the Youla parametrization.

One can also prove that for LFTs the Youla parametrization provides the same set that internally stabilizes the LFT loop. In order to prove this fact let us start from a double coprime factorization of $\mathrm{diag}\{0, K_0\}$. It turns out that, by inverting the usual roles, we have a dual Youla parametrization of P_g. It follows that P_g should have the following form

$$P_g = \begin{pmatrix} q_{zw} & q_{zu} \\ q_{yw} & 0 \end{pmatrix} \star \begin{pmatrix} -m^{-1}u & m^{-1} \\ \tilde{m}^{-1} & 0 \end{pmatrix} \star \begin{pmatrix} 0 & I \\ I & P_{yu} \end{pmatrix},$$

where q_{zw}, q_{zu}, q_{yw} are stable systems and $\star$ denotes the Redheffer (star) product

$$A \star B = \begin{pmatrix} \mathfrak{F}_l(A, B_{11}) & A_{12}(I - B_{11}A_{22})^{-1}B_{12} \\ B_{21}(I - A_{22}B_{11})^{-1}A_{21} & \mathfrak{F}_u(B, A_{22}) \end{pmatrix}.$$

The resulting closed–loop form for a stabilizing controller is given by

$$\mathfrak{F}_l(P_g, K) = q_{zw} + q_{zu}qq_{yw}, \tag{18}$$

where q is the Youla parameter of K relative to the given double coprime factorization of P_{yu}. As we have already emphasised all these results are also valid in the LPV framework.

A more advanced classical LTI application is the derivation of the suboptimal $\mathcal{H}_\infty$ controller set, see, e.g., Tsai and Gu (2014). The direct analogues of the J-unitary/outer factorizations computationally are not feasible in the LPV context. Nevertheless the so called J-negative/outer factorizations are applicable. Due to space limitations the topic cannot be elaborated further in this paper.

While the basic $\mathcal{H}_\infty$ controller design problem is tractable by using LMI techniques there is room for further improvements. Despite the fact that LPV design has been applied successfully for more then a decade, fundamental problems, e.g., tight estimation for the induced $\mathcal{L}_2$ gain, are still waiting for a solution.

Robust LPV design algorithms are quite involved, in general. We emphasise that besides the development of the design algorithms loop transformation tools might also facilitate the conceptual understanding of these methods. Moreover, as we have already seen, there is an intimate relationship between state and loop transforms. Both Lyapunov and IQC based approaches to robust design problems can be cast as transformations that allow to put the given problem in a particular advantageous form whose solution is almost trivial. This fact motivates our interest in the study of the loop transform in the LPV framework.

6. CONCLUSION

In this paper we have revisited some facts related to LPV models and LPV modelling. We focused on two different aspects of the topic: the state transformation, as a tool that relates equivalent descriptions of the same system and the loop transformation, which is based on an I/O view and concerns the preservation of stability of the closed loop.

The main goal of the paper was to provide an initialization of the novices in LPV modelling to obtain a concentrate view of the topic and in order to eliminate the possible pitfalls that still often occur in the related literature. Moreover, our intention was to point out some new research topics related to these transformation techniques that might also be interesting for a much larger audience. One of them was related to the lack of causality of the transformed systems if the transformation is parameter varying. The related practical question targets the possibility of the elimination of certain parameters (parameter derivatives) from the description of the system by applying suitable transformations.

REFERENCES

Alkhoury, Z., Petreczky, M., and Mercere, G. (2017). Comparing global input-output behavior of frozen-equivalent LPV state-space models. *20th IFAC World Congress Toulouse, IFAC-PapersOnLine*, 50(1), 9766 – 9771.

Anderson, B.D., Ilchmann, A., and Wirth, F.R. (2013). Stabilizability of linear time-varying systems. *Systems & Control Letters*, 62(9), 747–755.

Balas, G., Bokor, J., and Szabó, Z. (2003). Invariant subspaces for LPV systems and their applications. *IEEE Transactions on Automatic Control*, 48(11), 2065–2069.

Ball, J.A., Helton, J.W., and Verma, M. (1991). A factorization principle for stabilization of linear control systems. *Journal of Nonlinear and Robust Control*, 1, 229–294.

Blanchini, F., Casagrande, D., and Miani, S. (2010). Stable LPV realization of parametric transfer functions and its application to gain-scheduling control design. *IEEE Transactions on Automatic Control*, 55(10), 2271–2281.

Bokor, J. and Balas, G. (2004). Detection filter design for LPV systems—a geometric approach. *Automatica*, 40(3), 511–518.

Dewilde, P. and van der Veen, A.J. (1998). *Time-Varying State Space Realizations. In: Time-Varying Systems and Computations*, 33–72. Springer US, Boston, MA.

Feintuch, A. (1998). *Robust Control Theory in Hilbert Space*. Applied Math. Sciences 130. Springer, New York.

Isidori, A. and Ruberti, A. (1976). State-space representation and realization of time-varying linear input—output functions. *Journal of the Franklin Institute*, 301(6), 573–592.

Kalman, R.E. (1963). Mathematical description of linear dynamical systems. *SIAM Journal on Control*, 1(2), 152–192.

Kamen, E. (1979). New results in realization theory for linear time-varying analytic systems. *IEEE Transactions on Automatic Control*, 24(6), 866–878.

Luspay, T., Péni, T., Gőzse, I., Szabó, Z., and Vanek, B. (2018). Model reduction for LPV systems based on approximate modal decomposition. *Int. Journal for Numerical Methods in Engineering*, 113(6), 891–909.

Péni, T., Vanek, B., Lipták, G., Szabó, Z., and Bokor, J. (2017). Nullspace-based input reconfiguration architecture for overactuated aerial vehicles. *IEEE Transactions on Control Systems Technology*, PP(99), 1–8.

Petreczky, M., Tóth, R., and Mercere, G.. (2017). Realization theory for LPV state-space representations with affine dependence. *IEEE Transactions on Automatic Control*, 62(9), 4667–4674.

Silverman, L.M. and Meadows, H.E. (1969). Equivalent realizations of linear systems. *SIAM Journal on Applied Mathematics*, 17(2), 393–408.

Silverman, L.M. (1966). Representation and realization of time-variable linear systems. Technical report, Columbia Univ New York, Dept. of Electrical Eng.

Sontag, E. (1979). Realization theory of discrete-time nonlinear systems: Part I-The bounded case. *IEEE Transactions on Circuits and Systems*, 26(5), 342–356.

Szabó, Z., Bokor, J., and Balas, G. (2003). Inversion of LPV systems and its application to fault detection. In *Proc. of the 5th IFAC Symposium on fault detection supervision and safety for technical processes (SAFEPROCESS'03), Washington, USA*, 235 – 240.

Szabó, Z., Péni, T., and Bokor, J. (2015). Null-space computation for qLPV systems. *1st IFAC Workshop on Linear Parameter Varying Systems, Grenoble, France*, IFAC-PapersOnLine, 48(26), 170–175.

Szabó, Z., Seiler, P., and Bokor, J. (2017). Internal stability and loop-transformations: an overview on LFTs, Möbius transforms and chain scattering. *20th IFAC World Congress Toulouse, France, IFAC-PapersOnLine*, 50(1), 7547–7553.

Tóth, R. (2010). *Identification and modelling of linear parameter-varying systems*, vol. 403 of *Lecture Notes in Control and Information Sciences*. Springer, Heidelberg.

Tóth, R., Abbas, H.S., and Werner, H. (2012). On the state-space realization of LPV input-output models: Practical approaches. *IEEE Transactions on Control Systems Technology*, 20(1), 139–153.

Tóth, R., Felici, F., Heuberger, P.S.C., and den Hof, P.M.J.V. (2007). Discrete time LPV I/O and state space representations, differences of behavior and pitfalls of interpolation. In *2007 European Control Conference (ECC)*, 5418–5425.

Tsai, M.C. and Gu, D. (2014). *Robust and Optimal Control – A Two-port Framework Approach*. Springer-Verlag.

Vidyasagar, M. (1985). *Control System Synthesis: A Factorization Approach*. MIT Press, Cambridge.

Xie, W. and Eisaka, T. (2004). Design of LPV control systems based on Youla parameterization. *IEE Proceedings-Control Theory and Applications*, 151(4), 465–472.

Appendix A. LPV I/O VS. STATE SPACE

Let us consider the LPV plant $(P(\rho), \mathbf{\Omega})$ and the controller $(K(\rho), \mathbf{\Omega})$ through one of their particular state space representation as

$$P(\rho) \sim \left[\begin{array}{c|c} A(\rho) & B(\rho) \\ \hline C(\rho) & 0 \end{array}\right], \quad \text{and} \quad K(\rho) \sim \left[\begin{array}{c|c} A_K(\rho) & B_K(\rho) \\ \hline C_K(\rho) & 0 \end{array}\right],$$

i.e.,

$$\begin{aligned} \dot{x}(t) &= A(\rho)x(t) + B(\rho)u(t), \quad x(0) = 0, \\ y_P(t) &= C(\rho)x(t), \end{aligned}$$

and

$$\begin{aligned} \dot{x}_K(t) &= A_K(\rho)x_K(t) + B_K(\rho)y(t), \quad x_K(0) = 0, \\ u_K(t) &= C_K(\rho)x_K(t) + D_K(\rho)y(t), \end{aligned}$$

respectively.

Recall that for an invertible D and any Δ we have

$$\mathfrak{F}_u(\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right], \Delta)^{-1} = \mathfrak{F}_u(M_i, \Delta),$$

where

$$M_i = \left[\begin{array}{c|c} A - BD_a^{-1}C & BD_a^{-1} \\ \hline -D_a^{-1}C & D_a^{-1} \end{array}\right],$$

as an easy consequence of the matrix inversion lemma.

Then

$$\begin{pmatrix} I & K \\ P & I \end{pmatrix}(\rho) \quad \text{and} \quad \begin{pmatrix} I & K \\ P & I \end{pmatrix}^{-1}(\rho)$$

are LPV systems, represented through $\rho \in \mathbf{\Omega}$ and

$$\begin{aligned} \dot{\xi} &= \begin{pmatrix} A(\rho) & 0 \\ 0 & A_K(\rho) \end{pmatrix} \xi + \begin{pmatrix} B(\rho) & 0 \\ 0 & B_K(\rho) \end{pmatrix} z, \ \xi(0) = 0, \\ w &= \begin{pmatrix} 0 & C_K(\rho) \\ C(\rho) & 0 \end{pmatrix} \xi + \begin{pmatrix} I & D_K(\rho) \\ 0 & I \end{pmatrix} z, \end{aligned}$$

for the notations see Figure 1(a), and

$$\begin{aligned} \dot{\eta} &= \begin{pmatrix} A(\rho) + BD_kC(\rho) & -BC_K(\rho) \\ -B_KC(\rho) & A_K(\rho) \end{pmatrix} \eta + \begin{pmatrix} B(\rho) & -BD_K(\rho) \\ 0 & B_K(\rho) \end{pmatrix} w \\ z &= \begin{pmatrix} D_GC(\rho) & -C_K(\rho) \\ -C(\rho) & 0 \end{pmatrix} \eta + \begin{pmatrix} I & -D_K(\rho) \\ 0 & I \end{pmatrix} w, \ \eta(0) = 0, \end{aligned}$$

respectively.

Then, it is also straightforward to identify the state space representations for the block elements that appear in (11).