Iterative learning control and feedforward for LPV systems : applied to a position-dependent motion system

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Iterative Learning Control and Feedforward for LPV Systems: Applied to a Position-Dependent Motion System

Robin de Rozario, Tom Oomen and Maarten Steinbuch

Abstract— Iterative Learning Control (ILC) enables performance improvement by learning from previous tasks. The aim of this paper is to develop an ILC approach for Linear Parameter Varying (LPV) systems to enable improved performance and increased convergence speed compared to the linear time-invariant approach. This is achieved through dedicated analysis and norm-optimal synthesis of LPV learning filters. Application to a position-dependent motion system shows a significant improvement in accuracy and convergence rate, thereby confirming the potential of the proposed approach.

I. INTRODUCTION

Iterative Learning Control (ILC) enables the performance enhancement of systems that perform batch-to-batch tasks [1] such as: wafer stages [2], industrial printing systems [3], and pick-and-place machines [4].

Among the different ILC design frameworks, Linear Time-Invariant (LTI) methods [5] have gained popularity because accurate LTI models can be obtained using well-established system identification tools [6]. Using these models to design the ILC learning-filters leads to fast converging schemes, where the rate of convergence is governed by the model accuracy [7]. However, increasingly stringent throughput requirements result in ever lighter mechatronic systems [8] with configuration-dependent dynamics [9]. Since these systems cannot be reasonably approximated by an LTI model, excessive robustness measures are required in the ILC design to achieve convergence [10], or alternatively, the learning speed is decreased to the point where it takes many trials to converge to an acceptable level of performance [11]. These nonlinear dynamic systems can often be accurately modeled as Linear Parameter Varying (LPV) systems [12], as is evidenced by the large number of successful applications to mechatronic systems [13], [14], [15].

Extending the ILC framework to the class of LPV systems would greatly benefit these applications, as is recognized in [16], [17] and [10]. In [16], a D-type learning algorithm is proposed for Linear Time-Varying (LTV) systems by discretizing the time axis, which results in a controller that interpolates local controllers in the time domain. In [17], the scheduling parameter is assumed to be constant during each trial, but changes from trial-to-trial. This leads to a trial-varying learning filter that aims to optimize the performance despite trial-varying dynamics, but is not treating varying dynamics during a task. In [10], an LTV learning filter

Authors are with the Faculty of Mechanical Engineering, Control Systems Technology, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands. E-mail of corresponding author: `r.d.rozario@tue.nl`

is determined separately for each scheduling trajectory of interest, thereby not addressing the full LPV case.

Although recent developments have shown the potential of beyond-LTI learning filters in ILC, at present no systematic framework is available yet. The aim of this paper is to fill this gap through optimal discrete-time LPV filter design, using dedicated criteria that are derived based on ILC properties. By considering tasks of infinite time, an approach results wherein robustness can be explicitly addressed and which requires only a single synthesis step, thereby providing a significant extension to [10]. In addition, since feedforward control can be interpreted as a single ILC iteration, the same approach is applied to the synthesis of feedforward controllers. This is achieved through the following contributions.

C1 In Section II, the LPV ILC problem is posed and an update law is proposed along with conditions under which monotonic convergence is guaranteed. This results in an optimal left- and right-inverse matching problem for the design of learning filters and feedforward controllers respectively.

C2 In Section III, the structure of the inverse of LPV systems is analyzed and the obtained insights are used to extend fixed-lag smoothing [18] and preview-based control [19] to the LPV case, thereby effectively constituting a complete design framework for the synthesis of learning filters and feedforward controllers for discrete-time LPV systems.

C3 In Section IV, the proposed approach is applied to a simulated position-dependent motion system, thereby establishing the practical relevance of this method.

C2 is provides an extension of optimal inverse matching for LPV systems [20] by including preview to allow for non-causal filters. This is related to preview-based stable inversion [21] for Non-Minimum Phase systems (NMP). In this context, C2 extends stable inversion to LPV systems, by combining optimal inverse design with ILC as in [22].

A. Concepts and Notation

Let $\mathbb{R}$, $\mathbb{Z}$ and $\mathbb{Z}_+$ denote the set of reals, integers and nonnegative integers respectively. A discrete time signal s denotes the map $s : \mathbb{Z} \mapsto \mathbb{R}^{n_s}$, and q is defined as the shift operator, such that $q^\tau s[k] = s[k+\tau]$, with $\tau, k \in \mathbb{Z}$. This paper considers Single-Input Single-Output (SISO) discrete time LPV systems, which are represented in state-space as [23],

$$\Sigma(\rho) : \begin{cases} x[k+1] = A(\rho[k])x[k] + B(\rho[k])u[k], & \text{(1a)} \\ y[k] = C(\rho[k])x[k] + D(\rho[k])u[k], & \text{(1b)} \end{cases}$$

where $\rho[k] \in \mathcal{D} \subset \mathbb{R}^{n_\rho}$, with $\mathcal{D}$ the parameter space. For a given sequence ρ, the input-output map is denoted as $\Sigma(\rho) : u[k] \mapsto y[k]$, i.e. $y[k] = \Sigma(\rho[k])u[k]$, where the dependence on k is omitted when no ambiguity occurs. For a constant scheduling sequence, i.e. $\rho[k] = \bar{\rho} \in \mathcal{D} \ \forall k$, (1) describes an LTI system, which is called the frozen behavior of the LPV system at $\bar{\rho}$ [23]. Moreover, the l_2-induced norm of $\Sigma(\rho)$ for a given sequence ρ, is defined as,

$$\|\Sigma(\rho)\|_{2,2} \triangleq \sup_{u \in l_2} \frac{\|y\|_2}{\|u\|_2}, \quad \|s\|_2 = \sqrt{\sum_{k=-\infty}^{\infty} s^\top[k]s[k]}.$$

With slight abuse of notation, $\|.\|_{\mathscr{L}_\infty} \triangleq \|.\|_{2,2}$ and $\|.\|_{\mathscr{H}_\infty} \triangleq \|.\|_{2,2}$ indicate the $\mathscr{L}_\infty$ and $\mathscr{H}_\infty$-norm of operators operating on $l_2(\mathbb{Z})$ and $l_2(\mathbb{Z}_+)$, respectively. If $\Sigma(\rho)$ is such that $\|\Sigma(\rho)\|_{2,2} < \infty$ for every possible sequence ρ, the system is called Bounded-Input Bounded-Output (BIBO) stable in the l_2-norm [12]. In the remainder of this work, stability refers to BIBO stability. Moreover, let the plant to be controlled be given by the l_2-BIBO stable LPV system $G(\rho)$ with static parameter dependence, i.e. it depends only explicitly on $\rho[k]$ and not on $\rho[k+n]$, $n \in \mathbb{Z} \setminus 0$. Moreover, define the tracking error as $e \triangleq r - y$, where r is the desired output.

II. ILC AND FEEDFORWARD CONTROL OF LPV SYSTEMS BY INVERSE MATCHING

Iterative Learning Control is typically employed to improve the tracking performance of systems that perform repeating tasks. The aim of this section is to extend the ILC framework to LPV systems for trials of infinite length. To this end, an input update law is proposed and the corresponding convergence criterion is presented, which is constitutes contribution C1. The convergence condition leads to an optimal left-inverse matching problem for the ILC learning filter. In addition, it is shown that feedforward control leads to a similar right-inverse matching problem.

A. Iterative Learning Control for LPV Systems

For a system $G(\rho)$ with input u_j and scheduling sequence ρ_j, the tracking error during the j-th trial is given by,

$$e_j = r - G(\rho_j)u_j, \tag{2}$$

where the initial condition is assumed to be zero without loss of generality [7], i.e., $x[k_0] = 0$. ILC aims to decrease e_j by iteratively updating u_j, based on errors that were made during past trials. In the present paper, the following LPV ILC the update law is considered,

$$u_{j+1} = Q(u_j + L(\rho_j)e_j), \tag{3}$$

where Q is a stable, possibly non-causal LTI robustness filter and $L(\rho)$ is an LPV learning filter. This contains the LTI variant [5] as a special case, which is obtained by freezing the learning filter $L(\bar{\rho})$ for a specific value of the parameter. The benefit of the proposed extension with respect to the LTI case is shown in Section IV. The updated input generated by (3) can indeed increase the tracking performance iteratively as is presented in the following theorem.
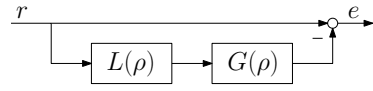

Fig. 1. $G(\rho)$ is the plant to be controlled by the feedforward controller $L(\rho)$ which the aims to reduce e, which is induced by r.

Theorem 1 (LPV ILC Convergence). *Given $G(\rho)$, a reference $r \in l_2$, and an admissible scheduling sequence ρ, which are both trial-invariant, i.e., $\rho_j = \rho_\infty \ \forall j$. Then, if,*

$$\|Q(1 - L(\rho_\infty)G(\rho_\infty))\|_{\mathscr{L}_\infty} < 1, \tag{4}$$

the sequences u_j and e_j, governed by (2) and (3), converge to the fixed point (u_∞, e_∞), given by,

$$u_\infty = [1 - Q(1 - G(\rho_\infty)L(\rho_\infty))]^{-1}QL(\rho_\infty)r, \tag{5}$$
$$e_\infty = r - G(\rho_\infty)u_\infty,$$

along all possible parameter trajectories ρ, where $\|u_j - u_\infty\|_2$ decreases to zero monotonically.

The proof follows by rewriting u_{j+1} in terms of r and u_j and determining the fixed point (5), which is enabled by assuming trial-invariant dynamics. Application of the Banach fixed-point theorem [7] to $\|u_j - u_\infty\|_2$ then results in (4).

The convergence criterion (4) can be interpreted as an inverse matching problem, since convergence is ensured if $L(\rho)$ accurately approximates $G^{-1}(\rho)$. In LTI methods, L is commonly constructed based on a model of G and the robustness filter Q is used to enforce convergence if the mismatch with the true inverse system is too large. Note that for $Q = 1$, it holds that $e_\infty = 0$ and perfect tracking is obtained if (4) is satisfied. In the remainder, it is assumed that $G(\rho)$ is exactly known and therefore $Q = 1$ considered throughout. Moreover, since the convergence speed is inversely proportional to the left hand side of (4) [7], fast convergence is achieved if $L(\rho)$ accurately approximates $G^{-1}(\rho)$. This aim is formalized as follows.

Problem 1 (Left-inverse matching). *Given $G(\rho)$, obtain,*

$$L^*(\rho) = \arg\min_{L(\rho)} \max_{\rho[k] \in \mathcal{D}} \|1 - L(\rho)G(\rho)\|_{\mathscr{L}_\infty}.$$

Since $L(\rho)$ is stable and possibly non-causal, existing LPV $\mathscr{H}_\infty$-synthesis approaches cannot be applied directly to solve Problem 1. In the next section, an extension of fixed lag smoothing [18] to LPV systems is considered to provide a suitable approach to Problem 1. First, it is shown that feedforward control leads to a similar right-inverse model matching problem.

B. Feedforward Control for LPV systems

In feedforward control, the system $G(\rho)$ is pre-compensated by a filter $L(\rho)$ to reduce the tracking as is shown in Figure 1. A norm-optimal approach to feedforward design for LPV systems, which allows for non-causal controllers, can be formulated as follows.

Problem 2 (Right-inverse matching). *Given $G(\rho)$, obtain,*

$$L^*(\rho) = \arg\min_{L(\rho)} \max_{\rho[k] \in \mathcal{D}} \|1 - G(\rho)L(\rho)\|_{\mathscr{L}_\infty}.$$

Note that in the SISO LTI case, Problem 1 and 2 are identical. However, for LPV systems this is not the case, since the commutativity property is lost. The implication of this difference will become apparent in the next section, where a solution is provided by extending the concept of preview control [19] to LPV systems.

III. OPTIMAL PREVIEW-BASED LPV INVERSES

The aim of this section is to obtain a method for solving Problem 1 and 2. First, an analytic solution is treated, which is only applicable to a limited class of systems. Extending this direct approach to general systems is hampered by two challenges. These are addressed by proposing an optimal $\mathcal{H}_\infty$-synthesis approach with four key recommendations on the structure of $L(\rho)$ that follow from the presented analysis. These formulations constitute contribution C2.

A. Inverting LPV Systems of Relative Degree Zero

Note that Problem 1 and 2 are solved for $L(\rho) = G^{-1}(\rho)$. In this section, the realization of the inverse of LPV systems, with relative degree zero, is considered to analyze the structure of the optimal solution $L^*(\rho)$. To this end, the relative degree is defined as follows.

Definition 1 (Relative degree). *A system $\Sigma(\rho)$ is said to have relative degree $\kappa \in \mathbb{N}$, where κ is the lowest number such that $u[k]$ appears explicitly in $y[k + \kappa]$ for some admissible ρ.*

In the analysis to come, it is assumed that the relative degree is constant for all admissible ρ. For systems $G(\rho)$ with $\kappa = 0$, $G^{-1}(\rho)$ allows the following state-space realization.

Lemma 1. *Let $G(\rho) : u \mapsto y$ be represented by (1) with $\kappa = 0$. Then, $G^{-1}(\rho) : y \mapsto u$ is given by,*

$$G^{-1}(\rho) : \left[\begin{array}{c|c} A(\rho) - B(\rho)D^{-1}(\rho)C(\rho) & B(\rho)D^{-1}(\rho) \\ \hline -D^{-1}(\rho)C(\rho) & D^{-1}(\rho) \end{array} \right].$$

The proof follows from inverting (1b) and substituting it in (1a). In practice, this direct approach provides a solution only for a limited class of systems due to the following.

I1 If $\kappa > 0$, $G^{-1}(\rho)$ is non-causal and cannot be described by a state-space realization as given by (1).

I2 This approach does not guarantee that $L(\rho)$ is stable. More specifically, if the frozen system $G(\bar{\rho})$ is NMP, $G^{-1}(\bar{\rho})$ is unstable, an hence $G^{-1}(\rho)$ is unstable.

Challenge I1 is tackled in the next subsection by shifting the input or output of the system forward in time, which is also known as previewing. In the subsection after that, this preview-based approach is combined with $\mathcal{H}_\infty$-optimization to provide a complete solution to Problem 1 and 2.

B. Towards Relative Degree Zero

Challenge I1 can be overcome as follows. I: Apply κ samples input- or output preview to $G(\rho)$ to obtain $\bar{G}(\rho) = G(\rho)q^\tau$ or $\bar{G}(\rho) = q^\tau G(\rho)$, respectively. II: Determine $\bar{L}(\rho) = \bar{G}^{-1}(\rho)$ by means of Lemma 1. III: Apply κ samples output- or input preview to $\bar{L}(\rho)$ respectively, by composing the non-causal filter as $L(\rho) = \bar{L}(\rho)q^\kappa$ or $L(\rho) =$

$q^\kappa \bar{L}(\rho)$, respectively. Note that the non-commutative nature of q^τ with LPV systems implies that when input-preview is applied to $G(\rho)$, output-preview needs to be applied to $\bar{L}(\rho)$ to maintain equivalence, and vice versa. Step I can be performed using state-space manipulations, as is presented in the following Lemma, whereas the filter $L(\rho)$ that results from step III does not allow a state-space representation since it is non-causal.

Lemma 2 (Input- and output-previewing). *Let $G(\rho[k]) : u[k] \mapsto y[k]$ be a system with relative degree κ. Then, $\bar{G}(\rho) : u[k+\kappa] \mapsto y[k]$, whose input is previewed by κ samples with respect to the input of $G(\rho)$, is given by,*

$$\bar{G}(\rho) : \left[\begin{array}{c|c} A(\rho[k]) & A^\kappa(\rho[k])B(\rho[k]) \\ \hline C(\rho[k]) & C(\rho[k])A^{\kappa-1}(\rho[k])B(\rho[k]) \end{array} \right].$$

Similarly, the realization of the output-previewed system $\bar{G}(\rho) : u[k] \mapsto y[k+\kappa]$, is given by,

$$\bar{G}(\rho) : \left[\begin{array}{c|c} A(\rho[k]) & B(\rho[k]) \\ \hline \bar{C}(\rho[k],...,\rho[k+\kappa]) & \bar{D}(\rho[k],...,\rho[k+\kappa]) \end{array} \right],$$

$$\bar{C}(\rho) = C(\rho[k+\kappa]) \prod_{j=0}^{\kappa-1} A(\rho[k+j]),$$

$$\bar{D}(\rho) = C(\rho[k+\kappa]) \prod_{j=1}^{\kappa-1} A(\rho[k+j])B(\rho[k]).$$

The proof follows by applying the appropriate state transformation to $\bar{G}(\rho)$ in the case of input-previewing, as will be presented elsewhere. The output-previewing case follows directly from recursive evaluation and redefining the output.

Only if $\bar{L}(\rho)$ is stable will this direct approach solve Problem 1 and 2. In the next section, stability of $\bar{L}(\rho)$ is ensured through an $\mathcal{H}_\infty$-synthesis approach that includes preview to enable non-causal solutions $L(\rho)$ in the same manner as presented in this section. In doing so, Lemma 2 provides insight on the structure of the optimal solution $L^*(\rho) = G^{-1}(\rho)$. Namely, when input previewing is applied, $\bar{G}(\rho)$ only depends on $\rho[k]$, whereas for output-previewing, $\bar{G}(\rho)$ also depends on its time shifts $\rho[k+1],...,\rho[k+\kappa]$. These insight are exploited in next section to provide the structure of $L(\rho)$ in the optimization approach.

C. Optimal Preview-based Iterative Learning Control

The direct preview-based approach only provides a solution for a limited class of systems, since it does not address I2. For Problem 1, this is overcome by recognizing that by using input-previewing, the reference can equivalently be delayed as is shown in Figure 2.II. This formulation is especially useful since both $\bar{G}(\rho)$ and $\bar{L}(\rho)$ are causal, and consequently, an $\mathcal{H}_\infty$ problem can be formulated, whose solution results in a stable controller $\bar{L}(\rho)$.

Problem 3 (Left-inverse matching with preview). *Given $G(\rho)$ and a number of preview samples τ, obtain,*

$$\bar{L}^*(\rho) = \arg\min_{\bar{L}(\rho)} \max_{\rho[k]\in\mathcal{D}} \|q^{-\tau} - \bar{L}(\rho)G(\rho)\|_{\mathcal{H}_\infty}. \quad (6)$$
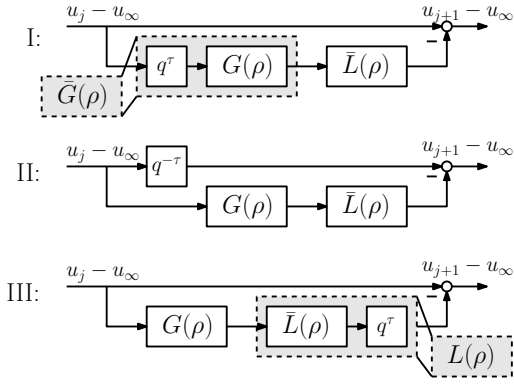
Fig. 2. I: For $G(\rho)$ with relative degree κ, the input is previewed by $\tau = \kappa$ samples, thereby making it biproper. Consequently, the inverse $\bar{L}(\rho)$ also has relative degree 0. II: This is equivalent to delaying the upper path, which enables $\mathscr{H}_\infty$-synthesis techniques to computing $\bar{L}(\rho)$. Additional preview can be added to compensate NMP dynamics, i.e., $\tau \geq \kappa$. III: The non-causal filter $L(\rho)$ is obtained by previewing the output of $\bar{L}(\rho)$.

Fig. 3. I: For $G(\rho)$ with relative degree κ, the output is previewed by $\tau \geq \kappa$ samples, thereby making it biproper. Consequently, the inverse $\bar{L}(\rho)$ also has relative degree 0. II: This is equivalent to delaying the upper path, which enables $\mathscr{H}_\infty$-synthesis techniques to computing $\bar{L}(\rho)$. Additional preview can be added to compensate NMP dynamics, i.e., $\tau \geq \kappa$. III: The non-causal filter $L(\rho)$ is obtained by previewing the input of $\bar{L}(\rho)$.

This problem can be solved using existing techniques [24], [25], [26] and this approach thereby overcomes I1 and I2 related to the direct approach. The complete solution to Problem 1 is depicted in Figure 2, where all three schemes are equivalent mappings. In Figure 2.II, the causal filter $\bar{L}(\rho)$ is obtained by solving Problem 3, whereas in Figure 2.III the non-causal filter $L(\rho)$ is obtained by output-previewing $\bar{L}(\rho)$. Realizing that the structure of $\bar{L}(\rho)$ is revealed by applying Lemma 1 to the realization of Lemma 2, leads to the following recommendations for synthesizing learning filters for LPV systems by means of solving Problem 3.

R1 It is sufficient for $\bar{L}(\rho)$ to depend explicitly on $\rho[k]$ only in order to include the case of perfect tracking.

R2 Care should be taken that $\bar{L}(\rho)$ allows a sufficiently rich description of the functional dependence of ρ to match $G^{-1}(\rho)$. For example, when $G(\rho)$ depends on ρ in an affine or polytopic fashion, $\bar{G}^{-1}(\rho)$ generally depends on ρ in a rational way. Consequently, methods for solving Problem 3 that allow rational dependence in $\bar{L}(\rho)$ are recommended, e.g. Linear Fractional Representation (LFR) synthesis tools, [24], [26].

R3 The number of preview samples should be $\tau \geq \kappa$, where κ is the relative degree of $G(\rho)$ to include the case of perfect tracking. Taking $\tau \gg \kappa$ may be necessary when $G(\bar{\rho})$ has NMP dynamics for some $\bar{\rho} \in \mathcal{D}$, in order to guarantee a satisfying solution.

D. Optimal Preview-based Feedforward Control

Along the same lines as in the previous subsection, issue I2 is overcome for Problem 2 by means of preview-control, which results in the following formulation.

Problem 4 (Right-inverse matching with preview). *Given $G(\rho)$ and a number of preview samples τ, obtain,*

$$\bar{L}^*(\rho) = \arg\min_{\bar{L}(\rho)} \max_{\rho[k] \in \mathcal{D}} \|q^{-\tau} - G(\rho)\bar{L}(\rho)\|_{\mathscr{H}_\infty}. \quad (7)$$

In this case, the reference can only be equivalently delayed by applying output-previewing as is shown in Figure 3. Consequently by Lemma 2, the optimal solution $\bar{L}^*(\rho)$
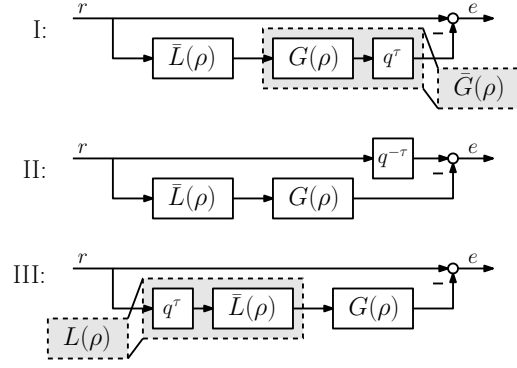
depends on $\rho[k], ..., \rho[k + \tau]$. This leads to the following recommendation for synthesizing Feedforward controllers of LPV systems by means of Problem 4.

R4 In addition to recommendations R2 and R3, $\bar{L}(\rho)$ should depend explicitly on $\rho[k], ..., \rho[k + \tau]$ in order to include the case of perfect tracking. This can be achieved by using parameter-dependent Lyapunov function-based synthesis [27].

In the aim of this section is to solve Problem 1 and 2. This is achieved by the formulation of Problem 3 and 4 in combination with recommendations R1-R4, which followed from analyzing the inverse of LPV systems. These approaches can be viewed as the LPV extensions of fixed-lag smoothing and preview control respectively.

IV. SIMULATION EXAMPLE

In this section, ILC and feedforward control are applied to an example system with position-dependent dynamics by applying the procedure as outlined in Section III to obtain the non-causal filters $\bar{L}(\rho)$. A comparison is made with an LTI approach which shows the relevance of this work and constitutes contribution C3.

A. Plant Description

Consider the positioning system as shown in Figure 5. Accurate positioning of the carriage in the y-direction is hampered by a resonance mode that varies as a function of its x-position as is schematically shown in Figure 5. A symmetric force is applied to beam $u = u_l = u_r$ and the output is the average of the collocated sensors, $y = \frac{1}{2}(y_l + y_r)$. A simplified polytopic LPV model with $\kappa = 1$ is given by,

$$G(\rho) : \left[\begin{array}{c|c} (1 - \rho[k])A_1 + \rho[k]A_2 & B \\ \hline C & 0 \end{array} \right], \quad \mathcal{D} = [0, 1],$$

$$A_1 = \begin{bmatrix} 0.99 & 0 & -0.001 & 0 \\ 0 & 0.99 & 0 & -0.001 \\ 0 & 0 & 0.99 & 0 \\ 0 & 3.969 & 0 & 0.9901 \end{bmatrix}, \quad B = \begin{bmatrix} -0.05 \\ 0 \\ 50 \\ 50 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0.9875 & 0 & -0.001 & 0 \\ 0 & 0.9875 & 0 & -0.001 \\ 0 & 0 & 0.9875 & 0 \\ 0 & 6.2016 & 0 & 0.9876 \end{bmatrix}, \quad C^\top = \begin{bmatrix} -1 \\ \beta \\ 0 \\ 0.001 \end{bmatrix}.$$
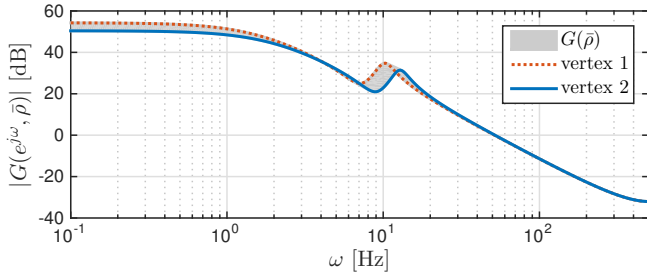
Fig. 4. The frozen dynamics $G(\bar{\rho})$ display a parameter-varying resonance.
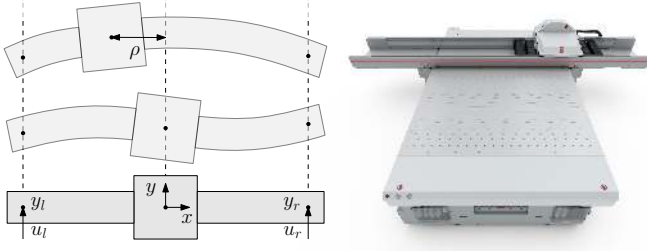

Fig. 5. The mode shapes of an industrial flat bed printer dependent on its configuration. A simplified model of this behavior is used here.

where for $\beta = 1$ in the C-matrix, the frozen behavior is shown in Figure 4, which resembles the dynamics of a position-dependent motion system, sampled at 1 kHz. The aim is to accurately perform the motion r, while the parameter ρ varies as shown in the upper plot of Figure 6.

B. Feedforward Control

Recommendations R2 and R4 as given in Section III-D state that $\bar{L}(\rho)$ should be adequately parametrized in terms of the dependency on ρ. The structure of the optimal $\bar{L}(\rho)$ is revealed by applying Lemma 2 to $G(\rho)$ with output preview for $\kappa = 1$, followed by applying Lemma 1. This results in,

$$\bar{G}(\rho): \left[\begin{array}{c|c} (1-\rho[k])A_1 + \rho[k]A_2 & B \\ \hline (1-\rho[k])CA_1 + \rho[k]CA_2 & CB \end{array} \right],$$

and subsequent application of Lemma 1 to $\bar{G}(\rho)$ yields,

$$\bar{G}^{-1}: \left[\begin{array}{c|c} (1-\rho[k])\hat{A}_1 + \rho[k]\hat{A}_2 & B(CB)^{-1} \\ \hline (CB)^{-1}C((1-\rho[k])A_1 + \rho[k]A_2) & (CB)^{-1} \end{array} \right],$$

$$\hat{A}_1 = [I - B(CB)^{-1}C]A_2, \quad \hat{A}_2 = [I - B(CB)^{-1}C]A_2.$$

It can thus be concluded that $\bar{G}^{-1}$ also depends on ρ in a polytopic fashion and it only depends explicitly on $\rho[k]$. For this case, Problem 4 is solved as presented in [25]. The obtained solution to problem 4 for $\tau = 1$, yields an upper bound on the $\mathcal{H}_\infty$-norm in (7) of $\gamma = 5.62 \cdot 10^{-4}$. Applying the obtained feedforward controller as shown in Figure 1, leads to u and e, as shown in the lower plot of Figure 6. This shows that the computed controller indeed achieves perfect tracking. Alternatively, the analytically determined $\bar{G}^{-1}(\rho)$ proves to be quadratically stable [13] and thus the direct approach yields the exact solution for this case.

C. Iterative Learning Control

Next, the system is considered for $\beta = -1$ which results in frozen NMP dynamics $\forall \bar{\rho} \in \mathcal{D}$. Consequently, the analytic
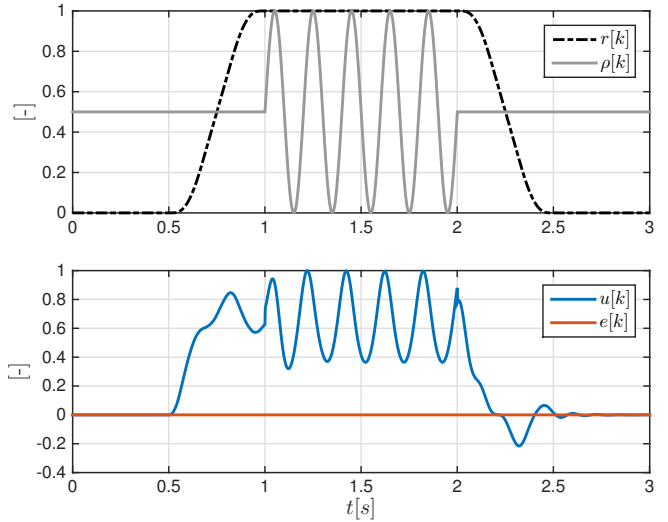

Fig. 6. Upper: the reference r and scheduling trajectory ρ. Lower: the optimal feedfoward input $u = L^\star(\rho)r$, which results in perfect tracking as is evidenced by the tracking error e. $L^\star(\rho)$ is obtained as in Section III.
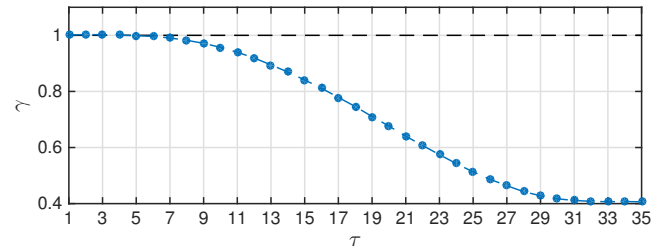

Fig. 7. The achievable performance bound γ as a function of the number of preview samples τ. This shows that the convergence rate can be increased by increasing τ, since γ is inversely proportional to this rate.

inverse is not stable and the procedure described in Section III-C is key to computing convergent learning filters. Along the same lines, it is concluded that a polytopic parametrization of $\bar{L}(\rho[k])$ is sufficient. Following recommendation R3, problem 3 is solved for increasing values of τ. Indeed, this results in an upper bound γ on the $\mathcal{H}_\infty$-norm in (6) that decreases as τ increases, as is shown in Figure 7. The obtained controller for $\tau = 35$ is used for ILC by updating the input as given by (3) with $Q = 1$. The progression of $\|e\|_2^2$ is shown as a function of the iterations in Figure 8. This shows that that $\|e\|_2^2$ converges to zero at a steady pace which results in excellent tracking performance after 20 iterations.

D. Comparison to ILC for LTI Systems

To highlight the relevance of this work, an LTI learning filter is implemented and compared to the LPV case. Since $\rho[k]$ averages around $\frac{1}{2}$, the LPV learning filter $L(\bar{\rho})$ is frozen at $\bar{\rho} = \frac{1}{2}$ to obtain a sensible LTI filter. It turns out that application of this filter does not lead to a convergent ILC due to the varying resonance. This can be visualized by considering the LTI convergence criterion [5] for frozen dynamics, i.e., considering (4), $\forall \bar{\rho} \in \mathcal{D}$, where $\rho[k] = \bar{\rho} \; \forall k$,

$$|Q(e^{j\omega})(1 - L(e^{j\omega}, \bar{\rho})G(e^{j\omega}, \bar{\rho}))| < 1, \quad \forall \omega \in [0, 2\pi). \quad (8)$$

This condition is necessary, but not sufficient in case $\rho[k]$ is time-varying. For the considered LTI filter with $Q = 1$,
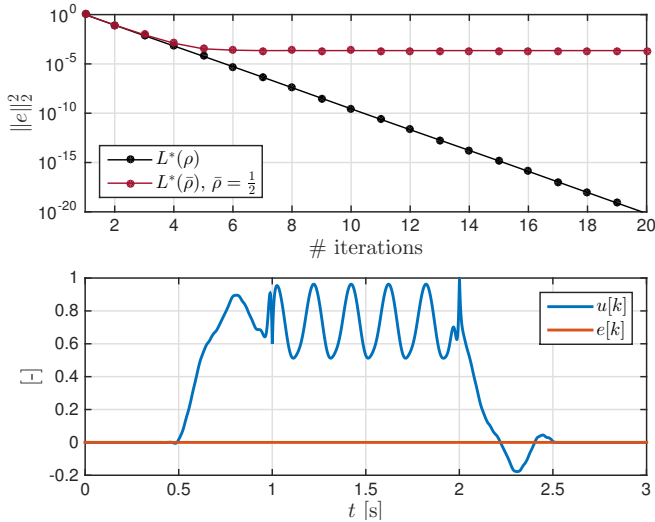
Fig. 8. Upper: $\|e\|_2^2$ converges for the LPV learning filter $L^\star(\rho)$, whereas for the LTI filter $L^\star(\bar\rho)$, with $\bar\rho = \frac{1}{2}$, additional robustness is required as is shown in Figure 9, resulting in limited performance. Lower: u and e that result after 20 iterations with the LPV filter, which show that perfect tracking is achieved.
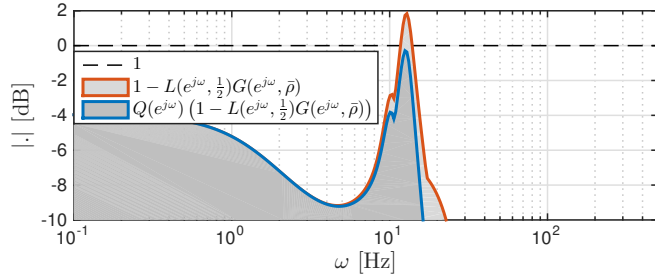

Fig. 9. Visualization of (8) for the LTI filter $L(\bar\rho)$ with $\bar\rho = \frac{1}{2}$. This shows that the criterion is violated due to the variation in resonant dynamics. To satisfy (8), robustness needs to be added, which is the main drawback of applying LTI techniques to LPV systems.

is (8) is violated as is shown in Figure 9. To satisfy (8), a robustness filter is added. Even though satisfying (8) is not sufficient, the ILC converges as is shown in Figure 8. However, the achieved performance is not as good as for the full LPV case. The latter did not require any robustness filter, since the parameter-varying dynamics are explicitly taking into account by the learning filter. This is the main motivation for adopting the full LPV method as proposed in this paper.

V. CONCLUSION

In this paper, an extension of ILC is proposed for discrete-time LPV systems. This is achieved by the development of a fixed-lag smoothing approach for the design of LPV learning filters. Moreover, a similar preview-based control approach is suggested for the design of LPV feedforward controllers. The effectiveness of these methods is shown by application to a motion system simulation study.

REFERENCES

[1] N. Amann, D. H. Owens, and E. Rogers, "Iterative learning control using optimal feedback and feedforward actions," *Int. J. Control*, vol. 65, no. 2, pp. 277–293, 1996.
[2] B. G. Dijkstra and O. H. Bosgra, "Extrapolation of optimal lifted system ILC solution, with application to a waferstage," in *Proc. Amer. Control Conf.*, vol. 4. IEEE, 2002, pp. 2595–2600.
[3] J. Bolder, T. Oomen, S. Koekebakker, and M. Steinbuch, "Using iterative learning control with basis functions to compensate medium deformation in a wide-format inkjet printer," *IFAC Mechatronics*, vol. 24, pp. 944–953, 2014.
[4] F. Boeren, A. Bareja, T. Kok, and T. Oomen, "Frequency-domain ILC approach for repeating and varying tasks: With application to semiconductor bonding equipment," *IEEE/ASME Trans. Mechatron.*, 2016.
[5] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Syst. Mag.*, vol. 26, no. 3, pp. 96–114, 2006.
[6] L. Ljung, "System identification: Theory for the user," *Englewood Cliffs*, 1987.
[7] K. L. Moore, *Iterative learning control for deterministic systems*. Springer-Verlag, 2012.
[8] T. Oomen, R. van Herpen, S. Quist, M. van de Wal, O. Bosgra, and M. Steinbuch, "Connecting system identification and robust control for next-generation motion control of a wafer stage," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 1, pp. 102–118, 2014.
[9] M. M. da Silva, W. Desmet, and H. V. Brussel, "Design of mechatronic systems with configuration-dependent dynamics: Simulation and optimization," *IEEE/ASME Trans. Mechatron.*, vol. 13, no. 6, pp. 638–646, 2008.
[10] J. van Zundert, J. Bolder, S. Koekebakker, and T. Oomen, "Resource-efficient ILC for LTI/LTV systems through LQ tracking and stable inversion: Enabling large feedforward tasks on a position-dependent printer," *IFAC Mechatronics*, vol. 38, pp. 76 – 90, 2016.
[11] Z. Bien and J.-X. Xu, *Iterative learning control: analysis, design, integration and applications*. Springer-Verlag, 2012.
[12] D. J. Leith and W. E. Leithead, "Survey of gain-scheduling analysis and design," *Int. J. Control*, vol. 73, no. 11, pp. 1001–1025, 2000.
[13] C. Hoffmann and H. Werner, "A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations," *IEEE Trans. Control Syst. Technol.*, vol. 23, pp. 416–433, 2015.
[14] M. Groot Wassink, M. van de Wal, C. Scherer, and O. Bosgra, "LPV control for a wafer stage: Beyond the theoretical solution," *Control Engineering Practice*, vol. 13, no. 2, pp. 231–245, 2005.
[15] M. Steinbuch, R. van de Molengraft, and A.-j. Van der Voort, "Experimental Modelling and LPV Control of a Motion System," in *Proc. Amer. Control Conf.*, Denver, Colorado, 2003, pp. 1374–1379.
[16] Z. G. Li, C. Wen, Y. C. Soh, and Y. Q. Chen, "Iterative Learning Control of Linear Parametrized Varying Uncertain Systems," in *Int. Conf. on Control, Automation, Robotics and Vision*, Singapore, 2000.
[17] M. Butcher and A. Karimi, "Linear Parameter-Varying Iterative Learning Control With Application to a Linear Motor System," *IEEE/ASME Trans. Mechatron.*, vol. 15, no. 3, pp. 412–420, 2010.
[18] L. Mirkin, "On the $\mathcal{H}_\infty$ fixed-lag smoothing: How to exploit the information preview," *Automatica*, vol. 39, no. 8, pp. 1495–1504, 2003.
[19] M. Tomizuka, "Optimal continuous finite preview problem," *IEEE Trans. Autom. Control*, vol. 20, no. 3, pp. 362–365, 1975.
[20] M. Sato, "Inverse system design for LPV systems using parameter-dependent Lyapunov functions," *Automatica*, vol. 44, pp. 1072–1077, 2008.
[21] Q. Zou, "Optimal preview-based stable-inversion for output tracking of nonminimum-phase linear systems," *Automatica*, vol. 45, no. 1, pp. 230–237, 2009.
[22] O. Markusson, H. Hjalmarsson, and M. Norrlof, "Iterative learning control of nonlinear non-minimum phase systems and its application to system and model inversion," in *Proc. IEEE Conf. Decision Control*, vol. 5, 2001, pp. 4481–4482.
[23] R. Tóth, *Modeling and identification of linear parameter-varying systems*. Springer-Verlag, 2010, vol. 403.
[24] P. Apkarian and P. Gahinet, "A convex characterization of gain-scheduled $\mathcal{H}_\infty$-controllers," *IEEE Trans. Autom. Control*, vol. 40, no. 5, pp. 853–864, 1995.
[25] P. Apkarian, P. Gahinet, and G. Beckers, "Self-scheduled control of linear parameter-varying systems: a design example," *Automatica*, vol. 31, pp. 1251–1261, 1995.
[26] C. W. Scherer, "LPV control and full block multipliers," *Automatica*, vol. 37, no. 3, pp. 361–375, 2001.
[27] F. Wu and K. Dong, "Gain-scheduling control of lft systems using parameter-dependent lyapunov functions," *Automatica*, vol. 42, no. 1, pp. 39–50, 2006.