Enhancing flatbed printer accuracy and throughput: optimal rational feedforward controller tuning via iterative learning control

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2613498, IEEE Transactions on Industrial Electronics

Enhancing Flatbed Printer Accuracy and Throughput: Optimal Rational Feedforward Controller Tuning via Iterative Learning Control

Joost Bolder, Jurgen van Zundert, Sjirk Koekebakker, Tom Oomen, *Member*, *IEEE*

***Abstract*—Advanced control methods potentially enable performance improvements in printing systems for minor additional costs. The aim of this paper is to develop a control framework that is capable of delivering throughput and accuracy enhancements for an industrial flatbed inkjet printer. The proposed method involves iterative learning control with a rational feedforward parameterization to enable varying position references that are required for printing. Experimental results highlight the efficacy of the proposed method in a comparison with related pre-existing learning control approaches.**

***Index Terms*—Data-based control, extrapolation capabilities, iterative learning control, printer, rational feedforward.**

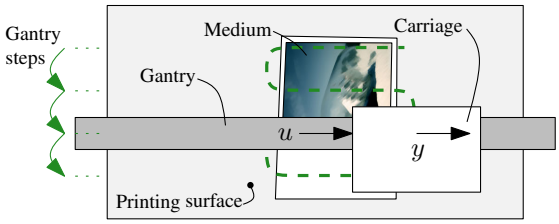

Fig. 1: Schematic overview of the flatbed inkjet printer (top view). An example movement of the carriage is indicated with the dashed line. The considered control variables are the carriage motor force u and carriage position y. Control of the gantry position is not considered in the present paper.

I. Introduction

A key business driver in the highly competitive printing industry is the cost/performance ratio of printing systems. The performance of a printing system is generally determined by the printing quality and throughput. Advanced control methods potentially enable performance improvements while keeping additional costs low, and are hence of valuable interest. Therefore, the focus in the present paper is on control of the motion system in an inkjet flatbed printer. Evidently, the market value of the printer depends on many mechatronic components and their control [1], including droplet generation [2], [3], and control of the medium conditioning [4].

An overview of the inkjet flatbed printer is considered in Fig. 1. The medium is placed on the printing surface and the carriage, which holds the printheads, moves over the medium while jetting droplets. This contrasts with more regular printers, where the medium is transported laterally to the printhead movements [5], [6]. In present commercial systems, throughput is sacrificed over print quality by introducing overlap in the printed passes to conceal printing errors [6]. Hence, achieving more accurate positioning can enable both higher quality and throughput.

The desired accuracy and throughput requirement impose high demands regarding the control specifications. The position references for the carriage are optimized for minimal printing time using the location of the medium and layout of the print. Hence, the references change each task, while it is important to maintain the accuracy level for the printing process. Employing standard feedback and feedforward controllers in this scenario generally results in moderate accuracy [7], which is robust for the changes in the tasks. On the other hand, Iterative Learning Control (ILC) [8]–[13] is known to achieve excellent accuracy in printing systems [3], [6], [14], [15], robotic applications [16], [17], and power systems [18]. In ILC, the control signals are updated from task to task by learning from measurement data of previous tasks. Learning control is applied to a planar 3D printer in [14], a planar ball-screw stage in [19], a flatbed inkjet printer in [20], and a piezoelectric stage in [21]. In spite of excellent reported tracking accuracy, the learned control signal is optimal for a fixed reference trajectory only. Since extrapolation to other references can lead to significant accuracy degradations [22], it is essential that extrapolation capabilities are introduced in ILC to achieve improvements in printing systems.

The excellent accuracy of ILC at the cost of poor extrapolation properties has led to the development of new learning-based algorithms that aim to combine the performance of ILC with the extrapolation properties of traditional feedforward controllers. Performance improvements of a flatbed printer using a model-based polynomial feedforward parameterization are presented in [23], of which the initial theoretical steps are

Manuscript received February 27, 2016; revised July 24, 2016; accepted August 29, 2016. This work is supported by Océ Technologies, P.O. Box 101, 5900 MA Venlo, The Netherlands. This work is also supported by the Innovational Research Incentives Scheme under the VENI grant “Precision Motion: Beyond the Nanometer” (no. 13073) awarded by NWO (The Netherlands Organization for Scientific Research) and STW (Dutch Science Foundation).

Joost Bolder, Jurgen van Zundert, and Tom Oomen are with the Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven 5600MB, The Netherlands (e-mail: jbolder@gmail.com; j.c.d.v.zundert@tue.nl; t.a.e.oomen@tue.nl).

Sjirk Koekebakker is with Océ Technologies B.V., Venlo, The Netherlands. (e-mail: sjirk.koekebakker@oce.com)

reported in [24], [25]. This polynomial parameterization is extended to rational parameterizations in [26] to further enhance the extrapolation capabilities and performance. Recently, it has been shown that these results are not fully optimal in the sense of minimizing a performance criterion, and in [27], a new ILC approach is developed that is proven to be optimal.

Although important progress in the control of printing systems is made, earlier results using a polynomial parameterization show a moderate accuracy [6], and do not fully achieve the potential performance. Therefore, the aim of the present paper is to exploit the recent advances in rational feedforward parameterizations and ILC algorithms in [27] to enable accuracy and throughput improvements in the flatbed printer. The main contribution of the present paper is the design and experimental implementation of a rational feedforward parameterization and iterative learning control in the industrial flatbed printer. In addition, the results in [27] are extended by adopting a stable inversion approach to enable a significant amount of preview through non-causal implementation. This potentially further improves performance. Comparison cases between the ILC approaches are presented using a benchmark study with experiments.

Conceptually different ILC approaches in order to introduce flexibility with respect to changing tasks have been developed to address related control problems in other applications. In [28], a segmented approach to ILC is presented and applied to a wafer stage. This approach is further extended in [15], where the complete task is divided into subtasks that are learned individually. The use of such a signal library is restricting in the sense that the tasks are required to consist of standardized building blocks that must be learned a priori. The use of a time-varying robustness filter [29], [30] introduces extrapolation capabilities for specific filter structures [30], though only for a restricted class of reference variations. In [31] an initial input selection for ILC is proposed. This method can be used to re-initialize the ILC after a reference change, see the related results in [32]. The re-initialization mapping is static and model-based, hence modeling errors directly affect the extrapolation capabilities. In addition, the reference must be known a priori. For a system-identification-based approach for the tuning of feedforward controllers, see [33]

This paper is organized as follows. First, control requirements for printing are derived in Section II. Next, the control approach is introduced in Section III. In Section IV, an ILC algorithm to determine the learning update with a rational feedforward parameterization is presented. In Section V, the experimental results are presented, followed by the conclusions in Section VI.

II. Control requirements for printing

In this section, control challenges for printing are illustrated using a state-of-the-art industrial flatbed printer. Several requirements for control are derived from these challenges. The goal of this paper is to develop a control framework that meets the requirements and to verify the designs using an experimental benchmark test on the flatbed printer.

A. Experimental system: Arizona flatbed printer

The industrial flatbed printer is an Arizona type 550GT [34] and is presented in Fig. 2. A schematic overview is presented in Fig. 1. The printer is normally used to print on rigid and flexible media such as cardboard, plastics, wood, acrylic glass, and metals. Typical applications include graphic arts, interior decoration, product decoration and signage. The maximum medium size is 2.5x1.25 m, and the maximum thickness is 50.8 mm. The carriage contains the printheads, which use piezoelectric inkjet printing technology. The ink is cured using UV lamps that are attached to the carriage.

To print an image, the carriage moves over the medium with constant velocity while the printheads deposit ink. After each pass of the carriage, the gantry steps in lateral direction while the carriage reverses. The stepping motion of the gantry, and scanning motion of the carriage are indicated in Fig. 1. The y position of the carriage is considered for control in the present paper. The stepping motion of the gantry does not limit the accuracy significantly and is therefore not considered. The carriage actuator is a current-driven brushless electrical motor with input u that is connected to the carriage via a steel belt-drive. The sensor is an optical encoder mounted inside the carriage, yielding a position measurement y.

A typical carriage movement consists of a constant velocity part, which is required by the inkjet printing process, and an acceleration part, where the carriage reverses direction. To minimize printing time, the reference trajectories for the carriage are tailored to the contents of the image to be printed. An example document is shown in Fig. 3. The example document consists of three text blocks which each decrease in width. To print this document in the shortest time, three carriage references are required that each decrease in distance.

The printer is connected to a control platform that is designed to create a large experimental freedom. The user interacts with a host PC. The host is connected to a target PC via a dedicated network interface. The target (see Fig. 2, top right) runs the real-time code and is connected to the sensors and actuators of the flatbed printer. The data acquisition hardware consists of a National Instruments (NI) PCI-6229 for digital and analog connections and a NI PCI-6602 for more digital connections. The carriage motor is a Moog brushless motor, which is driven by the Advanced Motion Controls B40A20I current amplifier. The power supply for the amplifiers is a Delta SM 70-45 with 3.2 kW power. The carriage position is measured with a Renishaw RGH41 encoder with 1 um resolution. The host computer runs Matlab/Simulink, where the simulink real-time toolbox is used to enable flexible control prototyping via user-friendly block diagrams. The main implementation aspects involve implementing non-causal filtering via stable inversion, which is explained in Section IV-B.

B. Control requirements

Developments in printing systems demand increased throughput and print quality. Achieving higher accuracy is essential in enabling both, as elaborated on in Section I. To improve the cost/performance ratio of printing systems, a

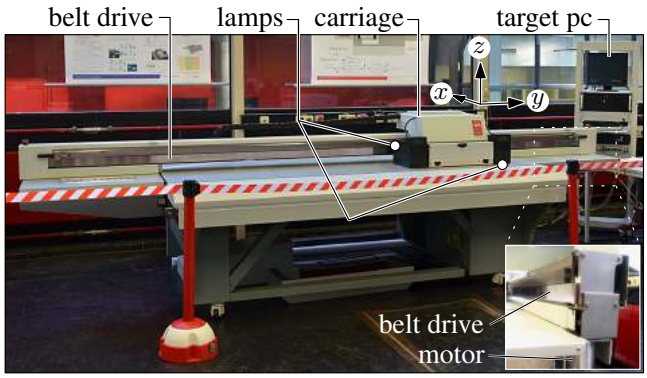

Fig. 2: Océ Arizona 550GT flatbed printer. The printer has four motion axes, the carriage can translate in the x, y, and z directions (indicated), and rotate around the z axis with a small angle φ (not indicated). The y position of the carriage is considered for control in the present paper. The position sensor for y is located inside the carriage, the motor with input u is connected via a belt-drive to the carriage. The printer is controlled with a real-time target computer (top right).

control method is required that enables highly accurate positioning, while maintaining flexibility with respect to changing references. The control method should therefore satisfy the following requirements.

R1 Accuracy: the control approach should achieve a high accuracy to enable high-quality prints and higher throughputs.

R2 Flexibility: to implement the control method in printing, the approach should be able to deal with varying reference trajectories.

The aim of this paper is to develop a control framework that addresses requirements R1 and R2. As illustrated in the previous section, the flatbed printer operates in a repeating fashion, executing similar tasks over and over. Employing standard feedback and feedforward controllers in this scenario generally results in moderate accuracy [7], which is robust for the changes in the tasks. Standard control approaches hence only satisfy R2, and not R1. This motivates the use of an ILC-based approach in view of R1, as it is well-known that ILC can deliver excellent performance in case the disturbances are trial-invariant [10]. To address R2, a suitable feedforward parametrization is introduced. It will be shown that when the parameterization encompasses the inverse system, perfect performance for any reference is achieved, satisfying both R1 and R2.

In practice, the presence of non-idealities including trial-varying disturbances, measurement noise, varying initial conditions, and position-dependent dynamics will affect the achievable performance. Therefore the control designs using this framework are validated in benchmark experiments on the Arizona flatbed printer. These benchmark tests are outlined next.

C. Benchmark test

In the benchmark experiments, the control designs are evaluated in their efficacy to address requirements R1 and R2, see Section II-B. Three reference trajectories for the carriage are applied consecutively, and the tracking performance is

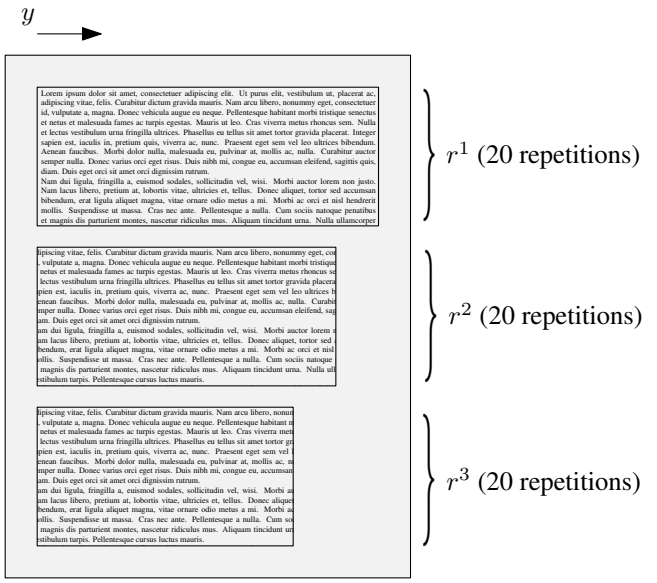

Fig. 3: Example of a printed document. The corresponding carriage position references are determined by the document contents, resulting in references r^1, r^2, and r^3.

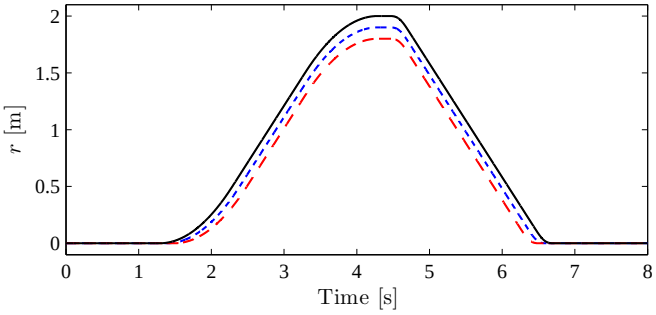

Fig. 4: Carriage position references r^1 (—), r^2 (---), and r^3 (- -).

evaluated. To print a high-quality image and meet requirement R2, it is essential that the tracking accuracy is maintained when switching references.

The references for the benchmark experiments correspond to the references required to print the example document in Fig. 3. The three trajectories are denoted as r^1, r^2, and r^3. The references are piecewise 4^{th} order polynomial functions [35], and are presented in Fig. 4. Forward and backward movements are performed, with a constant velocity part in between reversing direction.

The maximal displacement of r^1 equals 2 m, with a maximal velocity of 1 ms^{-1}, a forward acceleration of 1 ms^{-2}, and a backward acceleration of 6 ms^{-2}. The displacement of r^2 equals 1.9 m and of r^3 equals 1.8 m, with identical maximal accelerations and velocities as r^1.

The control framework is developed in the next section.

III. Rational feedforward parameterization

Throughout the paper, standard finite-time and frequency-domain representations for systems and signals are used [8]. This representation for signals and systems is general and captures physical models based on ordinary differential equations. The use is widely adopted in ILC, see, e.g., [10]. Without

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2613498, IEEE Transactions on Industrial Electronics

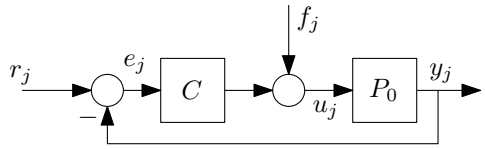

Fig. 5: Control setup.

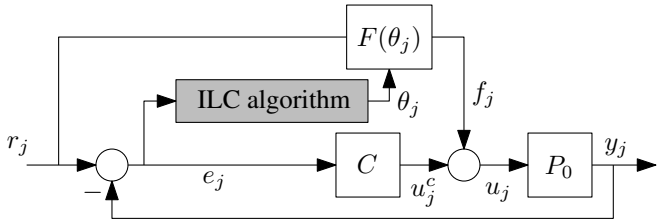

Fig. 6: Iterative learning control setup with basis functions $F(\theta_j)$ implemented.

explicit mentioning, systems and signals are in finite-time lifted notation, where the experiment length is N samples. A lifted system is denoted as P, where $P \in \mathbb{R}^{N\times N}$ is a Toeplitz matrix that contains impulse response coefficients of system P [10]. A lifted signal is denoted as u_j, where $u_j \in \mathbb{R}^{N\times 1}$ is an array of values. To indicate a frequency-domain representation of a system, an argument z is specified, e.g., $P(z)$, where z is a complex indeterminate.

A. Control setup

The control setup is shown in Fig. 5. All signals and systems are discrete-time. Each repetition of a reference is considered as a separate experiment or *trial* with a fixed length of N samples. The signals in Fig. 5 are indexed with trial index j. In Fig. 5, P_0 is the carriage with motor input u_j and position output y_j, C is a feedback controller, r_j is a trial-varying position reference, $e_j = r_j - y_j$ is the tracking error, and f_j is a feedforward signal. From Fig. 5 follows

$$e_j = S_0 r_j - S_0 P_0 f_j, \tag{1}$$

with $S_0 := (I + P_0 C)^{-1}$ the sensitivity. Notice that, theoretically, the error can be made arbitrarily small by minimizing S_0 through design of the feedback controller C. However, S_0 is subject to fundamental limitations [36], including the Bode sensitivity integral, and hence this is not a feasible solution for every r_j.

Feedforward is typically used to achieve a high performance. The idea is to determine f_j such that $e_j = 0$. From (1) results that setting $f_j = P_0^{-1} r_j$ leads to $e_j = 0$. The latter shows that this requires exact knowledge of the carriage dynamics P_0. In practice, P_0 is necessarily approximated using a model P, leading to performance deteriorations [37]. Hence, standard model-based feedforward is not directly suited in view of requirement R1, see Section II-B.

An alternative approach is to *learn* the signal f_j from measurement data of previous trials using a batch-to-batch control algorithm. This method is commonly referred to as Iterative Learning Control (ILC) [8]. While ILC is known to achieve excellent performance in practice, see e.g., [16]–[18], this requires a fixed reference $r_j = r,\ \forall j$. Changing the reference can lead to significant performance degradation, see [6], [32]. Hence, standard ILC is not directly suited in meeting requirement R2 in Section II-B.

To arrive at a control setup that potentially meets R1 and R2, the idea is to parameterize f_j in r_j using a filter. This is elaborated on next.

B. Rational feedforward parameterization

The basic idea in parameterizing f_j in r_j is that if r_j changes, also f_j is automatically adjusted. Let $f_j = F(\theta_j) r_j$, with $F(\theta_j)$ a filter with parameters θ_j. The particular structure of $F(\theta_j)$ and algorithms to determine θ_j remain to be chosen. Substitution of $f_j = F(\theta_j) r_j$ into (1) yields

$$\begin{aligned} e_j &= S_0 r_j - S_0 P_0 F(\theta_j) r_j \\ &= S_0 \left(I - P_0 F(\theta_j)\right) r_j. \end{aligned} \tag{2}$$

Equation (2) reveals that if the feedforward is parameterized in terms of the reference r_j, then the error in (2) can be made invariant under the choice of r_j, given that $F(\theta_j) = P_0^{-1}$. This yields perfect accuracy with $e_j = 0,\ \forall r_j$, potentially meeting requirements R1 and R2 in Section II. The introduced feedforward parameterization $F(\theta_j)$ is implemented in the controller structure and shown in Fig. 6.

As elaborated on earlier, to achieve a high accuracy $F(\theta_j) = P_0^{-1}$. The filter should capture the dynamics of P_0^{-1}. In the flatbed printer, ultraviolet (UV) light is used to cure the ink. Two UV lamps are connected to the carriage, these are indicated in Fig. 2. The mechanical decoupling of the lamps introduces complex conjugated poles and zeros in P_0, also known as (anti) resonances. Therefore, to achieve a high accuracy, $F(\theta_j)$ should also include poles and zeros. To this end, a rational structure for $F(\theta_j)$ is proposed. Let

$$F(\theta_j, z) = \frac{A(\theta_j, z)}{B(\theta_j, z)} \tag{3}$$

with

$$A(\theta_j, z) = \xi_0^A(z) + \sum_{n=1}^{m} \xi_n^A(z)\theta_j[n], \tag{4}$$

$$B(\theta_j, z) = \xi_0^B(z) + \sum_{n=1}^{m} \xi_n^B(z)\theta_j[n], \tag{5}$$

here, $\xi_n^A(z),\ \xi_n^B(z),\ n = 0, 1, 2, \ldots, m$ are polynomial transfer functions in complex indeterminate z, and $A(\theta_j, z)$ and $B(\theta_j, z)$ are linear in the parameters $\theta_j \in \mathbb{R}^{m\times 1}$. The parameters are indexed with $\theta_j[n]$, where, e.g., $\theta_j[3]$ is the third parameter. For this structure $F(\theta_j, z)$ is a *rational* function in θ_j. In case $B(\theta_j, z) = 1$, then $F(\theta_j, z) = A(\theta_j, z)$ is a *polynomial* function in θ_j.

The rational basis in (3) enables a high accuracy and can also introduce *extrapolation capabilities* with respect to the iteration varying reference r_j by virtue of (2). The structure for $F(\theta_j)$ in (3) should encompass the inverse plant to enable extrapolation capabilities with respect to all references. Choosing a structure for $F(\theta_j)$ can be viewed as selecting a model structure in a system identification procedure [33]. The dynamics of the flatbed printer include poles and zeros

and motivate a rational parameterization. In case P_0 only includes poles, a polynomial structure could suffice, e.g., as used in [6]. A key advantage of the rational parametrization combined with the stable inversion algorithm explained later in Section IV-B is infinite pre- and post actuation with only minor additional parameters, while this typically requires a high order polynomial parameterization, see [38].

The relation between θ_j and e_j is complicated for such rational $F(\theta_j)$ in general, and prohibits a manual tuning of the parameters θ_j. To achieve a high accuracy, the idea is to devise a learning-based algorithm to determine θ_{j+1} using measurement data.

IV. ILC ALGORITHMS FOR RATIONAL BASIS FUNCTIONS

Norm-optimal ILC is an important class of ILC algorithms, where f_{j+1} is determined from the solution of an optimization problem using measurements of e_j and f_j, see [9], [39], [40]. Inspired by norm-optimal ILC, a similar performance criterion is presented next.

Definition 1 (Performance criterion)**.** *The performance criterion is given by*

$$\mathcal{J}(\theta_{j+1}) := \|\hat{e}_{j+1}(\theta_{j+1})\|^2_{W_e} + \|f_{j+1}(\theta_{j+1})\|^2_{W_f} + \|f_{j+1}(\theta_{j+1}) - f_j\|^2_{W_{\Delta f}}, \quad (6)$$

with $W_e, W_f, W_{\Delta f} \succeq 0$ (positive semi-definite) and $\hat{e}_{j+1} = e_j - SP(f_{j+1} - f_j)$. Here, SP is a model of S_0P_0 and $\|x\|_W$ denotes a weighted 2-norm of a vector x, and is defined as $||x||_W := \sqrt{x^T W x}$, where W is a weighting matrix.

Compared to norm-optimal ILC, the key difference in (6) is that the argument is θ_{j+1}, which normally is f_{j+1}. This is an essential difference, leading to very different optimization algorithms for the rational case. In (6), W_e, W_f, and $W_{\Delta f}$ are user-defined weighting matrices to specify accuracy and robustness objectives such as robustness with respect to model uncertainty (W_f) and sensitivity to trial varying disturbances ($W_{\Delta f}$). The weighting matrices can be tailored towards different motion tasks such as reference tracking, point-to-point tasks, residual vibration suppression, and rate-limiting. A typical choice for the printing task in the present paper is to emphasize the constant velocity part of a reference using a diagonal W_e with varying elements. Specific tuning of the weighting matrices is discussed further in the experiments presented in Section V.

The parameters that minimize (6) are denoted as θ^*_{j+1}, and are given by

$$\theta^*_{j+1} = \arg\min_{\theta_{j+1}} \mathcal{J}(\theta_{j+1}). \quad (7)$$

The notation θ^*_{j+1} is to explicitly distinguish between θ_{j+1} as a variable, and a set of numerical values θ^*_{j+1}.

Using the rational basis in (3), instead of a polynomial basis or standard norm-optimal ILC, introduces significant complexity in determining (7). The key reason is that in case of a polynomial basis, the basis is linear in θ_j, hence (6) is a quadratic function in θ_j. Determining θ^*_{j+1} using (7) can in this case be performed by a straight-forward analytic solution to (7). This analytic solution is generally not possible in the rational case, as the gradient of (7) contains several nonlinear terms in θ_j.

A. An ILC-based algorithm for rational feedforward tuning

Several attempts have been made to minimize (6) given (3). In [27], it is shown that the earlier solution [26] is suboptimal. Here, an alternative algorithm is used, as first derived in [27]. The algorithm is based on setting

$$\frac{\partial \mathcal{J}(\theta_{j+1})}{\partial \theta_{j+1}} = 0. \quad (8)$$

Notice that this equation is nonlinear in θ_{j+1}. The basic idea to solve (8) using an iterative scheme. The nonlinear terms in (8) are fixed in each numerical iteration using the results of the previous iteration. Hence, this approach resorts to iterations of analytic solutions.

Suppose a specific rational structure $F(\theta_j)$ has been designed by selecting a number of parameters θ_j (see m in (4) and (5)) and by selecting the corresponding basis functions $\xi^A_n(z)$, and $\xi^B_n(z)$ for $n = 0, 1, 2, ..., m$ in (4), (5). Using the filter $F(\theta_j)$, experiments can be performed following the implementation in Fig. 6. In the following, the ILC algorithm is presented that enables the computation of θ^*_{j+1} in (7).

Algorithm 2 (ILC with a rational basis)**.** *Given initial parameters θ^*_0. Then, set $j = k = 0$ and perform the following steps:*

1) *Perform a trial with $f_j = F(\theta^*_j)r_j$ and measure e_j,*
2) *Determine $L^{\langle k \rangle}$, $Q^{\langle k \rangle}$, and $R^{\langle k \rangle}$, see* (12) *the Appendix.*
3) *Determine*

$$\theta^{\langle k \rangle}_{j+1} = L^{\langle k \rangle} e_j + Q^{\langle k \rangle} f_j - R^{\langle k \rangle} r_j. \quad (9)$$

4) *Set $k \to k+1$ and repeatedly go back to 2 until a prespecified maximal number of iterations $\bar{k}$ is met resulting in $\theta^{\langle k=\bar{k} \rangle}_{j+1}$, then proceed to the next step.*
5) *Set $\theta^*_{j+1} = \theta^{\langle k=\bar{k} \rangle}_{j+1}$ and implement $f_{j+1} = F(\theta^*_{j+1})r_{j+1}$, set $j \to j+1$, $k = 0$ and go back to step 1.*

Expressions for the learning filter $L^{\langle k \rangle}$, robustness filter $Q^{\langle k \rangle}$ and direct feedthrough $R^{\langle k \rangle}$ are given in the appendix. Note that the stable inversion algorithm in Section IV-B is employed to compute a bounded solution in case a filtering operation with an unstable (or non-causal) $F(\theta_j)$ needs to be performed.

Clearly, the parameters for the next trial θ_{j+1} are based on error measurements e_j of the previous trial, see (9). The convergence of Algorithm 2 is experienced to be good [27], as is also observed in closely related algorithms in instrumental variable system identification. If the numerical iterations in steps 2-4 (indexed with k) converge, then θ^*_{j+1} is a minimizer of the performance criterion in Definition 1.

In case P_0 includes non-minimumphase zeros, P_0^{-1} is unstable. In this case, the feedforward filter $F(\theta_j)$ should also be unstable, potentially leading to an unbounded f_j. In the next section, this issue is addressed with a stable inversion approach, enabling a bounded time-domain solution of $f_j = F(\theta_j)r_j$.

B. Stable inversion

Notice that there is no guarantee that $F(\theta_j, z)$ is a stable filter. Indeed, if P_0 includes non-minimumphase zeros, then it is directly seen from (2) that the optimal $F(\theta_j^*, z)$ is an unstable filter. A key aspect of the proposed approach is that it can explicitly deal with unstable feedforward filters. In particular, in the case $F(\theta_j, z)$ in (3) is unstable, the filtering operations cannot be performed in the usual manner, since forward time-domain computation leads to unbounded results. Several approaches to calculate the filtered signals can be pursued, including: i) stable approximations, e.g., see [41]–[43], and ii) the stable inversion approach in [44]. In the latter, the filter is seen as a non-causal operator instead of an unstable one, see also [45, Section 1.5].

Since stable approximations can lead to significant performance deteriorations, the stable inversion approach is adopted, see [44], [46] for initial results and [38] for an experimental validation, and [47] for further extensions. In the following, the relevant steps for calculating $f = F(\theta, z)r$ with unstable $F(\theta, z)$ using the exact stable inversion technique are presented. Assume that $F(\theta, z)$ has no poles on the unit circle in the complex plane and that $F(\theta, z)$ is a proper transfer function.

1) Let $F(\theta, z)$ have the state space realization A, B, C, D:

$$\begin{aligned} x(t+1) &= Ax(t) + Br(t) \\ f(t+1) &= Cx(t) + Dr(t) \end{aligned}$$

2) Introduce the following state partitioning:

$$\begin{aligned} \begin{bmatrix} x^s(t+1) \\ x^u(t+1) \end{bmatrix} &= \begin{bmatrix} A^s & 0 \\ 0 & A^u \end{bmatrix} \begin{bmatrix} x^s(t) \\ x^u(t) \end{bmatrix} + \begin{bmatrix} B^s \\ B^u \end{bmatrix} r(t) \quad (10) \\ f(t) &= \begin{bmatrix} C^s & C^u \end{bmatrix} \begin{bmatrix} x^s(t) \\ x^u(t) \end{bmatrix} + Dr(t) \end{aligned}$$

with $\lambda(A^s) \subset \overline{\mathbb{D}}$ and $\lambda(A^u) \cap \overline{\mathbb{D}} = \emptyset$, i.e., the state space realization is partitioned with all stable poles of $F(\theta, z)$ contained in A^s and the unstable poles in A^u. The closed unit disk is denoted as $\overline{\mathbb{D}}$ and $\lambda(\cdot)$ is the set of eigenvalues.

3) Solve *forwards* in time with $x(0) = x_0^s$:

$$x^s(t+1) = A^s x^s(t) + B^s r(t),$$

solve *backwards* in time with $x(N-1) = x_N^u$:

$$x^u(t+1) = A^u x^u(t) + B^u r(t).$$

4) Compute $f(t) = \begin{bmatrix} C^s & C^u \end{bmatrix} \begin{bmatrix} x^s(t) \\ x^u(t) \end{bmatrix} + Dr(t)$

Here, x_0^s is the initial condition of the stable part of $F(\theta, z)$ and x_N^u is the final condition of the unstable part of $F(\theta, z)$. Then, the resulting two-point boundary value problem (10) can be solved forwards and backwards in time [47]. If the preview time is sufficiently large at the start of the interval, then the error induced by finite time aspects are typically sufficiently small.

V. Experimental results

In this section, the basis design, controller tuning and experimental results are presented. The goal is to perform a benchmark test and analyze control performance in view of the requirements R1 and R2 in Section II-B for the proposed rational feedforward parameterization and corresponding ILC algorithm. In addition, an experimental comparison with a pre-existing polynomial basis, and standard norm-optimal ILC is presented.

A. Controller design

The controller design encompasses the following steps, 1. system identification, 2. rational basis design, 3. derivation of a polynomial basis and standard ILC as special cases, and 4. selection of the weighting matrices. Each step is presented next.

Step 1, system identification: A frequency response measurement of the carriage system $y = P_0 u$ is performed, see Fig. 1. A Gaussian noise excitation signal is used with a sampling time $T_s = 1$ ms. A Von Hann window is used to deal with leakage effects. The non-parametric model P_{frf} is estimated using the procedure in [48, Chapter 3]. The frequency response measurement P_{frf} results from 150 averaged measurement blocks to reduce the variance. The obtained frequency resolution is 0.125 Hz. The parametric model $P(z)$ is estimated using an iterative identification procedure, as e.g. in [49]. The resulting state-space model for $P(z)$ is of 10^{th} order. The Bode diagrams of P_{frf} and $P(z)$ are shown in Fig. 7. The identified model $P(z)$ resembles relatively well with the measurement P_{frf} for frequencies up to approximately 150 Hz. Analysis of $P_{\text{frf}}(z)$ reveals an anti-resonance at 7 Hz, and a resonance at 7.75 Hz. This anti-resonance/resonance pair is associated with the UV-lamps highlighted in Fig. 2. There are two more pairs of anti-resonance/resonances at approximately 30 and 52 Hz. The resonance at 100 Hz is caused by the decoupling of the carriage from the steel drive belt. Two poles below 0.25 Hz are present, illustrated by the -40 dB/decade slope in the low-frequency range.

Step 2, rational basis design: Analysis of the frequency response of P_0 in the previous step revealed several poles and zeros, constituting 2 integrators, and three anti-resonance/resonance pairs and a resonance. To capture the main phenomena in P_0, the numerator $A(\theta_j, z)$ and denominator $B(\theta_j, z)$, see (3), are selected as follows

$$\begin{aligned} A(\theta_j, z) &= K(z) \left(1 + \left(\tfrac{z-1}{zT_s}\right) \theta_j[1] + \left(\tfrac{z-1}{zT_s}\right)^2 \theta_j[2] + \right. \\ &\qquad \left. \left(\tfrac{z-1}{zT_s}\right)^3 \theta_j[3] + \left(\tfrac{z-1}{zT_s}\right)^4 \theta_j[4] \right), \quad (11) \\ B(\theta_j, z) &= 1 + \left(\tfrac{z-1}{zT_s}\right) \theta_j[5] + \left(\tfrac{z-1}{zT_s}\right)^2 \theta_j[6] + \\ &\qquad \left(\tfrac{z-1}{zT_s}\right)^3 \theta_j[7] + \left(\tfrac{z-1}{zT_s}\right)^4 \theta_j[8], \end{aligned}$$

where $K(z)$ is given by

$$K(z) = \frac{\alpha(z-\beta)(z-1)}{z}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2613498, IEEE Transactions on Industrial Electronics

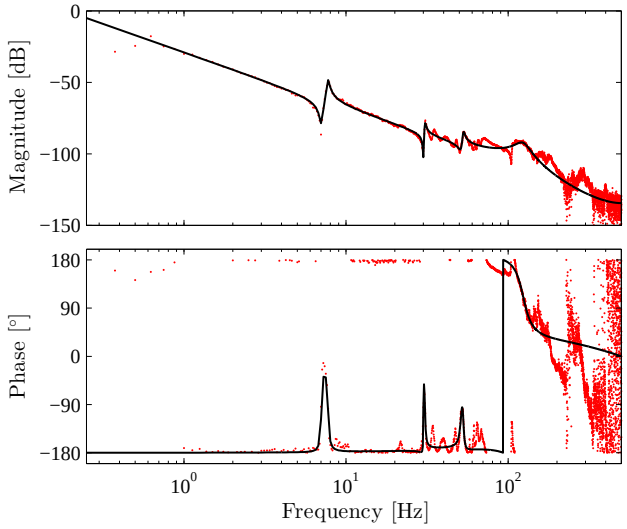

Fig. 7: Bode diagram of the frequency response measurement $P_{\text{frf}}(z)$ (···) and identified model $P(z)$ (−). The flexible dynamics of the UV lamps are clearly visible at approximately 7.5 Hz.

with, $\alpha = 6.40 \cdot 10^5$ and $\beta = 9.995 \cdot 10^{-1}$. The filter $K(z)$ constitutes a standard acceleration and viscous friction compensation, see [35]. This compensates for the two poles below 0.25 Hz in P_0. The polynomials $A(\theta_j, z)$ and $B(\theta_j, z)$ are of 4th order. This enables the compensation of two anti-resonance/resonance pairs.

The values for $\xi_n^A(z)$ and $\xi_n^B(z)$ for $n = 0, 1, ..., 8$ in (4) and (5) can be directly obtained from (11) and are not presented here for brevity.

Step 3, polynomial basis and standard ILC as special cases: The polynomial basis follows by setting $B(\theta_j, z) = 1$ in (11), yielding $f_j = A(\theta_j, z) r_j$. Hence, the structure of the polynomial basis is identical to the structure of the rational basis without the poles. Note that with $B(\theta_j, z) = 1$, the learning filters in Algorithm 2 are invariant over the iterations in k, hence step 4 in Algorithm 2 can be skipped and a direct analytic solution is obtained.

Standard norm-optimal ILC is recovered by setting $f_j = \theta_j$, with $\theta_j \in \mathbb{R}^N$, hence as many parameters as samples in a trial are used. The learning filters follow by setting $\Psi_{r_j}^A = I, \Psi_{e_j}^B = 0, \Psi_{f_j}^B = 0,\ \zeta^{\langle k \rangle} = I,\ \xi_0^A = 0$, and $\xi_0^B = 1$ in (12). Note that also in this case, step 4 in Algorithm 2 can be skipped, and the standard analytic ILC solution is obtained [9].

Step 4, selection of the weighting matrices: The weighting matrices in the performance criterion in (6) are selected $W_e = I \cdot 10^6$, and $W_f = W_{\Delta f} = 0$. This selection ensures that a minimal tracking error is pursued and that the learning rate is maximized. This enables a fair comparison between the different ILC approaches in view of the capable performance.

Remark 1 (Computational cost)**.** *In the rational case, it is experienced in practice that 7 to 10 iterations in k (see step 4 in Algorithm 2) is sufficient before proceeding to step 5. This increases the computational cost with respect to a polynomial parameterization with approximately a factor 10. The increased computational complexity does not hamper the application in the flatbed printer, since the parameter update is computed offline.*

B. Main experimental results

As indicated in Fig. 3, 60 trials are performed in total, where the reference r_j is given by

$$\begin{aligned} r_j &= r^1, \text{ for } 0 \leq j < 20, \\ r_j &= r^2, \text{ for } 20 \leq j < 40, \\ r_j &= r^3, \text{ for } 40 \leq j < 60. \end{aligned}$$

Hence, the reference is changed at trials 20 and 40. Requirement R1 in Section II-B involves achieving a high tracking performance, which means that the performance criterion $\mathcal{J}(\theta_j)$ should have a low value. Requirement R2 demands that this performance is maintained when the reference is changed. Hence, the change of the performance criterion at trials 20 and 40 is the measure for evaluating the control approaches in view of requirement R2. Note that depending on the printed image, the references may even change every trial, hence, much more often than the benchmark presented here.

The initial parameters θ_0 are set to values close to zero. For the rational basis functions, a maximum of 10 iterations in k is used, see step 4 in Algorithm 2. For standard ILC, $f_0 = 0$.

The algorithms are invoked, and the results are presented in Fig. 8, Fig. 9, and Fig. 10, revealing the following key observations.

Analysis of the performance criterion in Fig. 8 shows that the rational basis achieves a factor 3.3 better performance than the polynomial basis for r^1, a factor 1.9 for r^2 and a factor 1.15 for r^3. The performance of the ILC with a rational basis is much less sensitive to the reference variations than the performance of the polynomial basis and standard ILC, illustrating significantly improved extrapolation capabilities. For standard ILC, the value of the cost function increases from 0.12 to 19.7 at trial 20, and to similar values for trial 40. Standard ILC is hence not suited in view of requirement R2 in Section II-B. For the polynomial basis, the value decreases from 2.14 to 1.12 at trial 20 and increases from 1.14 to 1.67 at trial 40. For the rational parameterization, the performance remains at 0.67 at trial 20, and increases slightly from 0.66 to 1.1 at trial 40. Clearly, the rational parameterization is the best suited in view of meeting requirement R2. Although standard ILC achieves the lowest cost function values, the performance deteriorates drastically when the reference is changed, leading to unacceptable values. This will be particularly true if the reference changes more often or continuously. The performance in-between reference changes is more or less constant for all approaches. The level is likely caused by under-modeling in $F(\theta_j, z)$, which is a topic for future research.

Time-domain tracking errors are presented in Fig. 9, with a more detailed view of f_j and e_j from trial 10 in Fig. 10. The results show that when the reference changes, the tracking error with standard ILC increases dramatically. The results also show that the tracking error with the rational parameterization is smaller than with the polynomial one. Furthermore, the control signals of standard ILC and the rational parameterization clearly slow rational behavior, illustrated by the oscillatory nature. The polynomial parameterization cannot

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2016.2613498, IEEE Transactions on Industrial Electronics

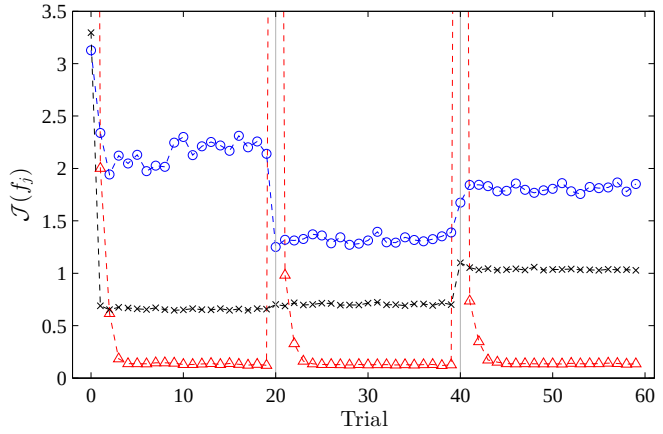

Fig. 8: Cost function values $\mathcal{J}(\theta_j)$. The performance of the rational basis functions ($\times$) is insensitive to the reference change $r^1 \rightarrow r^2$ at $j = 20$ and slightly sensitive at the reference change $r^2 \rightarrow r^3$ at $j = 40$. The performance with the polynomial basis ($\circ$) is more sensitive for both reference changes. The performance of standard norm-optimal ILC ($\triangle$) is extremely sensitive to the reference changes, resulting in a dramatic decrease in performance of a factor 159 for $r^1 \rightarrow r^2$ and a factor 165 for $r^2 \rightarrow r^3$.

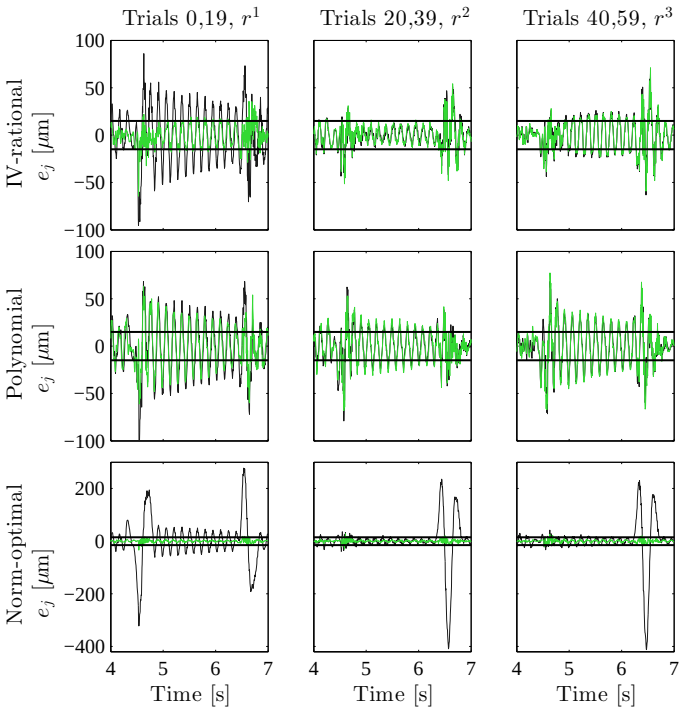

Fig. 9: Time-domain tracking errors: r^1 trial 0 (black) and 19 (green), (left column), r^2 trial 20 (black) and 39 (green), (middle column), r^3 trial 40 (black) and 59 (green), (right column), rational (top row), polynomial (middle row), and standard ILC (bottom row). As the reference changes, standard ILC shows a large error increase when compared with the parameterized approaches. The rational approach achieves the lowest racking errors.

generate control signals with infinite support, and results in a larger tracking error.

Concluding the experiments, the rational basis and Algorithm 2 perform well in view of requirements R1, and R2 in Section II, demonstrating significant accuracy improvements and flexibility with respect to reference changes. The polynomial basis performs worse than the rational basis in terms of both R1 and R2. Standard ILC achieves a very high accuracy, and is hence very suited for R1, but clearly lacks flexibility

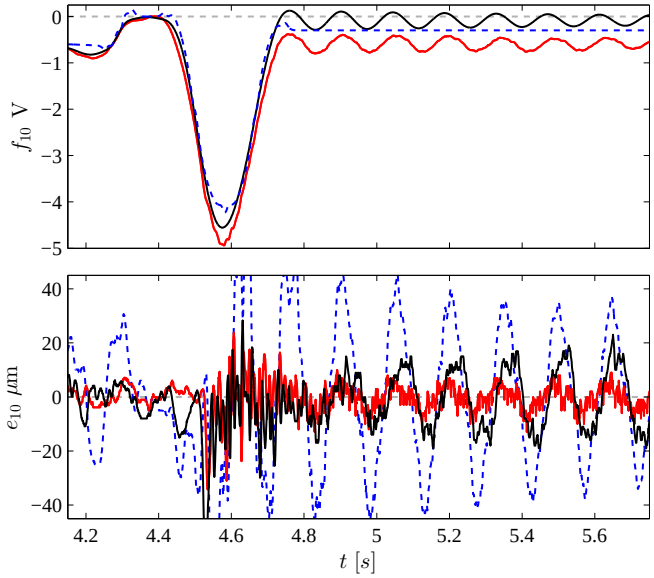

Fig. 10: Time domain measurements of f_{10} (top) and e_{10} (bottom) for the rational basis (—), polynomial basis (- -), and standard ILC (—). Standard ILC and ILC with a rational parameterization compensate the oscillations in the error to a large extent, in contrast to the polynomial parameterization, which cannot generate a signal with infinite support.

with respect to reference changes, and does not satisfy R2.

VI. Conclusion and Outlook

The main result of the present paper is a control approach to meet increasing throughput and accuracy demands for an industrial flatbed printer. Printing systems require a high positioning accuracy while maintaining a large freedom in the reference trajectories to achieve optimal throughput.

A rational feedforward parameterization and suitable ILC is employed. The rational basis can compensate effects related to parasitic flexible dynamics in the flatbed printer. The advantages of using a rational basis in ILC for a printing system are confirmed in a relevant experimental case study, where the proposed approach is compared with a pre-existing polynomial parameterization and standard ILC. Significant performance improvements as well as enhanced extrapolation capabilities are demonstrated.

Ongoing research is towards multivariable systems and position-dependent disturbances and dynamics.

Appendix

A detailed derivation of the learning filters is presented in [27]. The approach in [27] has close connections with commonly used algorithms for Instrumental Variable (IV) system identification [50].

If $k = 0$, then initialize $\theta_{j+1}^{\langle k-1 \rangle} = \theta_j$. The learning filters for rational basis functions with the IV approach are given by

$$\begin{aligned} L^{\langle k \rangle} &= \left(\zeta^{\langle k \rangle}\Gamma\right)^{-1} \zeta^{\langle k \rangle} J^T W_e \xi_0^B, \\ Q^{\langle k \rangle} &= \left(\zeta^{\langle k \rangle}\Gamma\right)^{-1} \zeta^{\langle k \rangle} \left(J^T W_e J + W_{\Delta f}\right) \xi_0^B, \\ R^{\langle k \rangle} &= \left(\zeta^{\langle k \rangle}\Gamma\right)^{-1} \zeta^{\langle k \rangle} \left(J^T W_e J + W_f + W_{\Delta f}\right) \xi_0^A, \end{aligned} \tag{12}$$

with

$$\Gamma = \left(J^T W_e J + W_f + W_{\Delta f}\right) \Psi^A_{r_j} - J^T W_e \Psi^B_{e_j} - \left(J^T W_e J + W_{\Delta f}\right) \Psi^B_{f_j}.$$

Here,

$$\zeta^{\langle k \rangle} = \left(\overline{\frac{\partial f^{\langle k-1 \rangle}_{j+1}}{\partial \theta^{\langle k-1 \rangle}_{j+1}}} \right)^T B^{-1}(\theta^{\langle k-1 \rangle}_{j+1}),$$

where

$$\overline{\frac{\partial f^{\langle k-1 \rangle}_{j+1}}{\partial \theta^{\langle k-1 \rangle}_{j+1}}} = B\left(\theta^{\langle k-1 \rangle}_{j+1}\right)^{-1} \Psi^A_{r_j} - A(\theta^{\langle k-1 \rangle}_{j+1}) B\left(\theta^{\langle k-1 \rangle}_{j+1}\right)^{-2} \Psi^B_{r_j}.$$

To compute the latter in case of unstable $B(\theta^{\langle k-1 \rangle}_{j+1})^{-1}$, the stable inversion approach in Section IV-B is used. Next, $J := SP$ is a model of the process sensitivity $S_0 P_0$. The basis functions in lifted notion are given by

$$\begin{aligned}
\Psi^A_{r_j} &= \left[\xi^A_1 r_j, \quad \xi^A_2 r_j, \quad \ldots, \quad \xi^A_m r_j\right], \\
\Psi^B_{e_j} &= \left[\xi^B_1 e_j, \quad \xi^B_2 e_j, \quad \ldots, \quad \xi^A_m e_j\right], \\
\Psi^B_{f_j} &= \left[\xi^B_1 f_j, \quad \xi^B_2 f_j, \quad \ldots, \quad \xi^B_m f_j\right], \\
\Psi^B_{r_j} &= \left[\xi^B_1 r_j, \quad \xi^B_2 r_j, \quad \ldots, \quad \xi^B_m r_j\right].
\end{aligned}$$

Here, ξ^A_n and ξ^B_n are the lifted representations corresponding with $\xi^A_n(z)$ and $\xi^A_n(z)$ in (4) and (5). The weighting matrices $W_e, W_f, W_{\Delta f} \succeq 0$ should be chosen such that (6) is a well-defined criterion, i.e., $\zeta^{\langle k \rangle}\Gamma$ must have full rank. It is experienced that this is often directly satisfied by setting $W_e \succ 0$ (positive definite) for the rational feedforward parameterization.

ACKNOWLEDGMENT

The authors would also like to acknowledge the fruitful discussions with Maarten Steinbuch. Bart Moris is acknowledged for performing preliminary experimental work.

REFERENCES

[1] S. Pond, *Inkjet Technology and Product Development Strategies*. Torrey Pines Research, 2000.

[2] A. Khalate, X. Bombois, G. Scorletti, R. Babuska, S. Koekebakker, and W. de Zeeuw, "A Waveform Design Method for a Piezo Inkjet Printhead Based on Robust Feedforward Control," *Journal of Microelectromechanical Systems*, vol. 21, pp. 1365–1374, Dec. 2012.

[3] M. G. Wassink, O. Bosgra, and S. Koekebakker, "Enhancing Inkjet Printhead Performance by MIMO Iterative Learning Control Using Implementation Based Basis Functions," *Proc. Americ. Contr. Conf., New York, NY, USA*, pp. 5472–5477, 2007.

[4] P. Zapata, M. Fransen, J. ten Thije Boonkkamp, and L. Saes, "Coupled Heat and Moisture Transport in Paper With Application to a Warm Print Surface," *Appl. Math. Model.*, vol. 37, pp. 7273–7286, Jul. 2013.

[5] J.-P. Gazeay, A. Eon, S. Zeghloul, and M. Arsicualt, "New Printing Robot for High-Resolution Pictures on Three-Dimensional Wide Surfaces," *IEEE Trans. Ind. Electr.*, vol. 58, pp. 384–391, Feb. 2011.

[6] 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," *Mechatronics*, vol. 24, pp. 944–953, Dec. 2014.

[7] M. Steinbuch, R. Merry, M. Boerlage, M. Ronde, and M. van de Molengraft, *Advanced Motion Control Design, Chapter 27 in the Control Systems Handbook - Second Edition*. CRC Press, Taylor & Francis Group, 2011.

[8] D. Bristow, M. Tharayil, and A. Alleyne, "A Survey of Iterative Learning Control: a Learning-based Method for High-performance Tracking Control," *IEEE Contr. Syst. Mag.*, vol. 26, pp. 96–114, Jun. 2006.

[9] S. Gunnarsson and M. Norrlöf, "On the design of ILC algorithms using optimization," *Automatica*, vol. 37, pp. 2011–2016, Dec. 2001.

[10] M. Norrlöf and S. Gunnarsson, "Time and Frequency Domain Convergence Properties in Iterative Learning Control," *Int. J. Contr.*, vol. 75, pp. 1114–1126, Sep. 2002.

[11] K. Abidi and J.-X. Xu, "Iterative Learning Control for Sampled-Data Systems: From Theory to Practice," *IEEE Trans. Ind. Electr.*, vol. 58, pp. 3002–3015, Jul. 2011.

[12] R.-E. Precup, S. Preitl, J. Tar, M. Tomescu, M. Takács, P. Korondi, and P. Baranyi, "Fuzzy Control System Performance Enhancement by Iterative Learning Control," *IEEE Trans. Ind. Electr.*, vol. 55, pp. 3461–3475, Sep. 2008.

[13] K.-S. Tzeng, D. Tzeng, and J.-S. Chen, "An Enhanced Iterative Learning Control Scheme Using Wavelet Transform," *IEEE Trans. Ind. Electr.*, vol. 52, pp. 922–924, Jun. 2005.

[14] K. Barton, D. Hoelzle, A. Alleyne, and A. Johnson, "Cross-Coupled Iterative Learning Control of Systems with Dissimilar Dynamics: Design and Implementation for Manufacturing Applications," *Int. J. Contr.*, vol. 84, pp. 1223–1233, Jul. 2011.

[15] D. Hoelzle, A. Alleyne, and A. Johnson, "Basis Task Approach to Iterative Learning Control With Applications to Micro-Robotic Deposition," *IEEE Trans. Contr. Syst. Techn.*, vol. 19, pp. 1138–1148, Sep. 2011.

[16] Y. Zhai, Y. Lin, F. Xi, and G. S, "Calibration-Based Iterative Learning Controlfor Path Tracking of Industrial Robots," *IEEE Trans. Ind. Electr.*, vol. 62, pp. 2921–2929, May. 2015.

[17] X. Li, Q. Ren, and J.-X. Xu, "Precise Speed Tracking Control of A Robotic Fish via Iterative Learning Control," *IEEE Trans. Ind. Electr.*, vol. 63, pp. 2221–2228, Apr. 2015.

[18] H. Deng, R. Oruganti, and D. Srinivasan, "Analysis and Design of Iterative Learning Control Strategies for UPS Inverters," *IEEE Trans. Ind. Electr.*, vol. 54, pp. 1739–1751, Jun. 2007.

[19] H. Fujimoto and T. Takemura, "High-Precision Control of Ball-Screw-Driven Stage Based on Repetitive Control Using n-Times Learning Filter," *IEEE Trans. Ind. Electr.*, vol. 61, pp. 3694–3703, Jul. 2014.

[20] J. Bolder, S. Kleinendorst, and T. Oomen, "Data-Driven Multivariable ILC: Enhanced Performance and Robustness by Eleminiating L and Q Filters," *Accepted, Int. J. Rob. Nonlin. Contr.*.

[21] D. Huang, J.-X. Xu, V. Venkataramanan, and T. Huynh, "High-Performance Tracking of Piezoelectric Positioning Stage Using Current-Cycle Iterative Learning Control With Gain Scheduling," *IEEE Trans. Ind. Electr.*, vol. 61, pp. 1085–1098, Feb. 2014.

[22] S. Gunnarsson and M. Norrlöf, "On the Disturbance Properties of High Order Iterative Learning Control Algorithms," *Automatica*, vol. 42, pp. 2031–2034, Nov. 2006.

[23] D. Kostic, P. van Zutven, P. Smulders, S. Koekebakker, and I. Smits, "Flatbed Printer Development - From Model-based Dynamics Analysis to Motion Control Performance Improvement," *Mikroniek*, vol. 5, p. 38, 2015.

[24] S. van der Meulen, R. Tousain, and O. Bosgra, "Fixed Structure Feedforward Controller Design Exploiting Iterative Trials: Application to a Wafer Stage and a Desktop Printer," *J. Dyn. Syst., Meas., and Contr.: Transactions of the ASME*, vol. 130, p. 16, Aug. 2008.

[25] J. van de Wijdeven and O. Bosgra, "Using Basis Functions in Iterative Learning Control: Analysis and Design Theory," *Int. J. Contr.*, vol. 83, pp. 661–675, Apr. 2010.

[26] J. Bolder and T. Oomen, "Rational Basis Functions in Iterative Learning Control - With Experimental Verification on a Motion System," *IEEE Trans. Contr. Syst. Techn.*, vol. 23, pp. 722–729, Mar. 2015.

[27] J. van Zundert, J. Bolder, and T. Oomen, "Optimality and Flexibility in Iterative Learning Control for Varying Tasks," *Automatica*, vol. 67, pp. 295–302, May. 2016.

[28] S. Mishra, J. Coaplen, and M. Tomizuka, "Precision Positioning of Wafer Scanners Segmented Iterative Learning Control for Nonrepetitive Disturbances," *IEEE Contr. Syst. Mag.*, vol. 27, pp. 20–25, Aug. 2007.

[29] D. Bristow and A. Alleyne, "Monotonic Convergence of Iterative Learning Control for Uncertain Systems Using a Time-varying Filter," *IEEE Trans. Automat. Contr.*, vol. 53, pp. 582–585, Mar. 2008.

[30] I. Rotariu, M. Steinbuch, and R. Ellenbroek, "Adaptive Iterative Learning Control for High Precision Motion Systems," *IEEE Trans. Contr. Syst. Techn.*, vol. 16, pp. 1075–1082, Sep. 2008.

[31] C. Freeman, M. A. Alsubaie, Z. Cai, E. Rogers, and P. Lewin, "Initial Input Selection for Iterative Learning Control," *J. Dyn. Syst., Meas., and Contr.*, vol. 133, p. 6, Aug. 2011.

[32] P. Janssens, G. Pipeleers, and J. Swevers, "Initialization Of ILC Based on a Previously Learned Trajectory," *Proc. Americ. Contr. Conf., Montréal, Canada*, pp. 610–614, 2012.

[33] F. Boeren, T. Oomen, and M. Steinbuch, "Iterative Motion Feedforward Tuning: A Data-driven Approach Based on Instrumental Variable Identification," *Contr. Eng. Prac.*, vol. 37, pp. 11–19, Apr. 2015.

[34] Océ Technologies B.V., "Arizona 550 GT/XT User Manual, Version 1.2 Revision B," pp. 1–269, 2012.

[35] P. Lambrechts, M. Boerlage, and M. Steinbuch, "Trajectory Planning and Feedforward Design for Electromechanical Motion Systems," *Contr. Eng. Prac.*, vol. 13, pp. 145–157, Feb. 2005.

[36] M. Seron, J. Braslavsky, and G. Goodwin, *Fundamental Limitations in Filtering and Control*. Springer-Verlag, London, 1997.

[37] S. Devasia, "Should Model-Based Inverse Inputs Be Used as Feedforward Under Plant Unvertainty?" *IEEE Trans. Automat. Contr.*, vol. 47, pp. 1865–1871, Nov. 2002.

[38] L. Blanken, F. Boeren, D. Bruijnen, and T. Oomen, "Rational Iterative Feedforward Tuning: Approaches, Stable Inversion, and Experimental Comparison," *Proc. Americ. Contr. Conf., Boston, MA, USA*, pp. 2629–2634, 2016.

[39] N. Amann, D. Owens, and E. Rogers, "Iterative Learning Control Using Optimal Feedback and Feedforward Actions," *Int. J. Contr.*, vol. 65, pp. 277–293, Feb. 1996.

[40] J. Lee, K. Lee, and W. Kim, "Model-based Iterative Learning Control with a Quadratic Criterion for Time-varying Linear Systems," *Automatica*, vol. 36, pp. 641–657, May. 2000.

[41] M. Tomizuka, "Zero Phase Error Tracking Algorithm for Digital Control," *J. Dyn. Syst., Meas., and Contr.*, vol. 109, pp. 65–68, May. 1987.

[42] E. Gross, M. Tomizuka, and W. Messner, "Cancellation of Discrete Time Unstable Zeros by Feedforward Control," *J. Dyn. Syst., Meas., and Contr.: Transactions of the ASME*, vol. 116, pp. 33–38, 1994.

[43] J. Butterworth, L. Pao, and D. Abramovitch, "Analysis and Comparison of Three Discrete-time Feedforward Model-inverse Control Techniques for Nonminimum-phase Systems," *Mechatronics*, vol. 22, pp. 577–587, Aug. 2012.

[44] S. Devasia, D. Chen, and B. Paden, "Nonlinear Inversion-Based Output Tracking," *IEEE Trans. Automat. Contr.*, vol. 41, pp. 930–942, Jul. 1996.

[45] G. Vinnicombe, *Uncertainty and Feedback: $\mathcal{H}_\infty$ loop-shaping and the ν-gap metric*. Imperial College Press, London, UK, 2001.

[46] T. Sogo, "On the Equivalence Between Stable Inversion for Nonminimum Phase Systems and Reciprocal Transfer Functions Defined by the Two-sided Laplace Transform," *Automatica*, vol. 46, pp. 122–126, Jan. 2010.

[47] 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," *Mechatronics*, vol. 38, pp. 76–90, Sep. 2016.

[48] R. Pintelon and J. Schoukens, *System Identification, a Frequency Domain Approach*. 2nd Edition, Wiley, 2012.

[49] R. van Herpen, T. Oomen, and M. Steinbuch, "Optimally Conditioned Instrumental Variable Approach for Frequency-domain System Identification," *Automatica*, vol. 50, pp. 2281–2291, Sep. 2014.

[50] M. Gilson, H. Garnier, P. Young, and P. Van den Hof, "Optimal instrumental variable method for closed-loop identification," *IET Control Theory & Applications*, vol. 5, no. 10, pp. 1147–1154, Jul. 2011.

Joost Bolder received the M.Sc. degree (cum laude) in Mechanical Engineering from the Eindhoven University of Technology, the Netherlands, in June 2011. In September 2015, he received the PhD degree from the Eindhoven University of Technology. Presently, Joost is a design engineer at ASML, Veldhoven, The Netherlands. His research interests include iterative learning control, image processing, and mechatronics.

Jurgen van Zundert received the M.Sc. degree (with great appreciation) in Mechanical Engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands in 2014. He is currently pursuing the Ph.D. degree in the Control Systems Technology group within the department of Mechanical Engineering at TU/e. His research interests include feedforward motion control, multi-rate control, and iterative learning control.

Sjirk Koekebakker received the M.Sc. and Ph.D. degrees in mechanical engineering from Delft University of Technology, Delft, The Netherlands, in 1993 and 2001, respectively. Since 1999, he is a Mechatronic Designer with Océ Technologies B.V., Venlo, The Netherlands, which is part of the Canon group. As of 2009, he has also been a part-time Senior Researcher in the Control Systems Technology Group, Mechanical Engineering Department, Eindhoven University of Technology, Eindhoven, The Netherlands. His main research interests are iterative learning and repetitive control, precision motion control, and printing systems.

Tom Oomen (M'06) received the M.Sc. degree (cum laude) and Ph.D. degree from the Eindhoven University of Technology, Eindhoven, The Netherlands. He held visiting positions at KTH, Stockholm, Sweden, and at The University of Newcastle, Australia. Presently, he is an assistant professor with the Department of Mechanical Engineering at the Eindhoven University of Technology. He is a recipient of the Corus Young Talent Graduation Award and the 2015 IEEE Transactions on Control Systems Technology Outstanding Paper Award. He is Associate Editor on the IEEE Conference Editorial Board and IFAC Mechatronics. His research interests are in the field of system identification, robust control, and learning control, with applications in mechatronic systems.