

## Integrated Rational Feedforward in Frequency-Domain Iterative Learning Control for Highly Task-Flexible Motion Control

Tsurumoto, Kentaro; Ohnishi, Wataru; Koseki, Takafumi; Van Haren, Max; Oomen, Tom

**DOI**

[10.1109/TMECH.2024.3400252](https://doi.org/10.1109/TMECH.2024.3400252)

**Publication date**

2024

**Document Version**

Final published version

**Published in**

IEEE/ASME Transactions on Mechatronics

**Citation (APA)**

Tsurumoto, K., Ohnishi, W., Koseki, T., Van Haren, M., & Oomen, T. (2024). Integrated Rational Feedforward in Frequency-Domain Iterative Learning Control for Highly Task-Flexible Motion Control. *IEEE/ASME Transactions on Mechatronics*, 29(4), 3010-3018.  
<https://doi.org/10.1109/TMECH.2024.3400252>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.





**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

# Integrated Rational Feedforward in Frequency-Domain Iterative Learning Control for Highly Task-Flexible Motion Control

Kentaro Tsurumoto , Wataru Ohnishi , Takafumi Koseki , Max van Haren , and Tom Oomen 

**Abstract**—Iterative learning control yields accurate feedforward input by utilizing experimental data from past iterations. However, typically there exists a tradeoff between task flexibility and tracking performance. This study aims to develop a learning framework with both high task-flexibility and high tracking-performance by integrating rational basis functions with frequency-domain learning. Rational basis functions enable the learning of system zeros, enhancing system representation compared to polynomial basis functions. The developed framework is validated through a two-mass motion system, showing high tracking-performance with high task-flexibility, enhanced by the rational basis functions effectively learning the flexible dynamics.

**Index Terms**—Basis functions, feedforward (FF) control, frequency-domain design, iterative learning control (ILC), stable inversion.

## I. INTRODUCTION

FEEDFORWARD (FF) control is essential for high-precision mechatronics systems where high speed and high precision are required, e.g., lithography systems [1], [2] and atomic force microscopy [3]. For typical model-based FF control, a precise and highly reliable FF is constructed based on the accurate system model [4]. However, as systems become more complex, the modeling effort increases [5].

Manuscript received 17 January 2024; revised 24 March 2024; accepted 3 May 2024. Date of publication 29 May 2024; date of current version 16 August 2024. Recommended by Technical Editor B. Chu and Senior Editor Q. Zou. This work was supported in part by the JSPS Bilateral Program under Grant JPJSBP220234403, in part by the JSPS Grant-in-Aid for Scientific Research (B) Program under Grant 23H01431, in part by the ECSEL Joint Undertaking under Grant 101007311 (IMOCO4.E), in part by the Netherlands Organisation for Scientific Research (NWO) within the Research Programme VIDI under Project 15698, and in part by the JSPS Grant-in-Aid for JSPS Fellows under Grant 24KJ0959. (Corresponding author: Kentaro Tsurumoto.)

Kentaro Tsurumoto, Wataru Ohnishi, and Takafumi Koseki are with the Department of Electrical Engineering and Information Systems, The University of Tokyo, Tokyo 113-8656, Japan (e-mail: k.tsurumoto@ctl.t.u-tokyo.ac.jp; ohnishi@ctl.t.u-tokyo.ac.jp; koseki@ctl.t.u-tokyo.ac.jp).

Max van Haren is with the Department of Mechanical Engineering, Eindhoven University of Technology, 5600MB Eindhoven, The Netherlands (e-mail: m.j.v.haren@tue.nl).

Tom Oomen is with the Department of Mechanical Engineering, Eindhoven University of Technology, 5600MB Eindhoven, The Netherlands, and also with the Faculty of Mechanical, Maritime, and Materials Engineering, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: t.a.e.oomen@tue.nl).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TMECH.2024.3400252>.

Digital Object Identifier 10.1109/TMECH.2024.3400252

Iterative learning control (ILC) [6], [7] is a data-based control method applied to improve the performance of systems with batchwise repetitive operations. By utilizing experimental data from past iterations, ILC learns an accurate FF input for the given operation. The primary features of ILC are high tracking-performance and analytical measures for assuring learning convergence. With an increasing demand for higher speed and accuracy, the following requirements are imposed on ILC to be applicable for next-generation industrial motion systems [5]:

- R1 high task-flexibility for nonrepetitive tasks;
- R2 high tracking-performance;
- R3 robust learning.

In this study, two types of ILC are considered and referred to as 1) frequency-domain ILC (F-ILC) [8], [9] and 2) basis function ILC (B-ILC) [10], [11].

F-ILC is an iterative learning method utilizing FF input update laws in frequency-domain system representation. F-ILC can achieve perfect tracking for systems only containing trial-invariant exogenous signals [9], satisfying requirement R2. In addition, requirement R3 is satisfied with convergent learning of F-ILC being verifiable before starting the experiment, using frequency response functions (FRFs), which are accurate and inexpensive to obtain [12]. However, as F-ILC is primarily targeted for optimizing a single repetitive task, changing the task requires reoptimization, violating requirement R1 [10].

B-ILC is an iterative parameter tuning method for the FF controller, utilizing a time-domain norm optimization problem. By using linear combinations of basis functions (e.g., velocity, acceleration, and snap) to construct the FF input, the dynamics of the system poles can be learned without being severely dependent on the exact same operation, partially satisfying all requirements R1–R3. Such linear selections are called polynomial basis functions [B-ILC (pol)] [2], [11] related to finite impulse response (FIR) filtering. However, typically due to basis functions being limited in expressivity, B-ILC cannot compose an FF input accurately as that of F-ILC, i.e., R1 is achieved with the compromise of R2.

To further relax the tradeoff between requirements R1 and R2, several developments have been made.

In rational B-ILC [B-ILC (rat)] [10], [13], [14] basis functions are combined nonlinearly to construct an FF controller with a rational structure. This enables learning of system zeros on top of system poles, resulting in improved task-flexibility (R1) and tracking performance (R2) than that of B-ILC (pol).

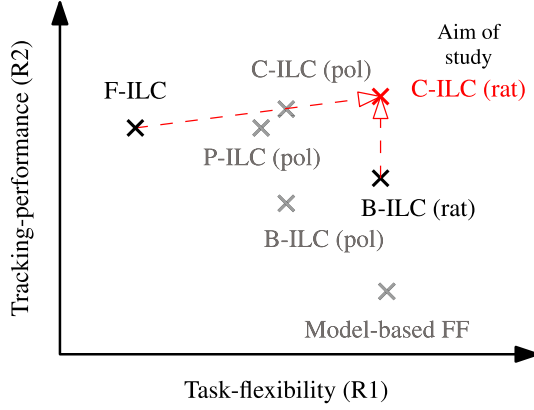


Fig. 1. Tradeoff between task flexibility (R1) and tracking performance (R2) for ILC [10]. In this study, C-ILC (rat) combined by frequency-domain design (F-ILC) and rational basis functions [B-ILC (rat)] is investigated.

In projection ILC (P-ILC) [15], the FF input learned by F-ILC is projected to polynomial basis functions enforcing flexibility against task variation. This enables partial satisfaction of requirement R1 while retaining the benefit of F-ILC (R2 and R3). In combined ILC (C-ILC) [16], B-ILC (pol) is integrated with F-ILC, and both frameworks are simultaneously learned. Not only does C-ILC retain both benefits of B-ILC (pol) and F-ILC, but it can potentially exceed the performance of F-ILC (R2).

However, for previous P-ILC and C-ILC frameworks, only polynomial basis functions have been implemented, limiting the task flexibility compared to that of B-ILC (rat).

An overview of the introduced ILC frameworks is summarized in Fig. 1.

Although rational basis functions can significantly enhance the task flexibility from polynomial basis functions, introducing it with frequency-domain learning comes with several complexities. First, unlike polynomial basis functions, rational basis functions are combined nonlinearly introducing a nonconvex optimization problem. Second, the nonconvex optimization scheme has to be developed individually for the ILC framework, considering the interference from the parallelly learned frequency-domain FF input.

The aim of this study is to develop a C-ILC framework incorporating rational basis functions, which can capture the system zero dynamics on top of pole dynamics for further enhanced task-flexibility (R1). The main contribution of this study is the proposal of integrating B-ILC (rat) with F-ILC, coupled with the following contributions.

- C1 Sequential rational basis function update law is developed for C-ILC (rat), avoiding interference from the parallelly learned frequency-domain FF input.
- C2 Convergence condition of C-ILC is formulated for ensuring robust learning (R3).
- C3 C-ILC (rat) is experimentally validated with a two-mass motion system, successfully fulfilling all requirements R1–R3.

A preliminary study focusing on the combined learning of F-ILC and B-ILC (pol) is reported in [16] with a simulation setup on a system with negligible system zeros. In this study, the task flexibility of C-ILC framework is fundamentally enhanced by incorporating rational basis functions, and results are experimentally verified on a system with considerable system zeros. In terms of novelty in theory, a sequential updating law for integrating rational basis functions in C-ILC is developed, and the convergence condition of C-ILC is formulated.

### A. Notation

Let  $\mathbf{H}(z)$  denote a discrete-time linear time invariant (LTI), single-input single-output (SISO) system. The FRF of  $\mathbf{H}(z)$  is obtained by substituting  $z = e^{i\omega} \forall \omega \in [0, 2\pi)$  and is denoted as  $\mathbf{H}(e^{i\omega})$ . Throughout this article,  $(z)$  is omitted, and  $\hat{\mathbf{H}}$  is a model of  $\mathbf{H}$ . The input and output of  $\mathbf{H}$  are  $u$  and  $y$ , and the signal length is assumed as  $N \in \mathbb{N}$ .

Let  $h(t)$  be the impulse response of  $\mathbf{H}$ . The finite-time convolution matrix  $H \in \mathbb{R}^{N \times N}$  corresponding to  $\mathbf{H}$  is expressed as

$$\underbrace{\begin{bmatrix} y[0] \\ y[1] \\ \vdots \\ y[N-1] \end{bmatrix}}_{\underline{y}} = \underbrace{\begin{bmatrix} h(0) & h(-1) & \cdots & h(1-N) \\ h(1) & h(0) & \cdots & h(2-N) \\ \vdots & \vdots & \ddots & \vdots \\ h(N-1) & h(N-2) & \cdots & h(0) \end{bmatrix}}_H \underbrace{\begin{bmatrix} u[0] \\ u[1] \\ \vdots \\ u[N-1] \end{bmatrix}}_{\underline{u}}$$

where  $\underline{u} \in \mathbb{R}^{N \times 1}$  and  $\underline{y} \in \mathbb{R}^{N \times 1}$  are  $u$  and  $y$ , respectively, lifted for  $N$  samples, assuming zero initial and final conditions.

Additionally, a two norm for vector  $x$  is given by  $\|x\|_2 = \sqrt{x^\top x}$  with its square being  $\|x\|^2 = x^\top x$ , and  $I_N$  is an  $N \times N$  identity matrix.

## II. PROBLEM FORMULATION

In this section, the considered problem is defined by describing the system and introducing model-based FF, F-ILC, and rational B-ILC.

### A. System Description

The control setup is shown in Fig. 2. Here, system  $\mathbf{G}$  and FB controller  $\mathbf{K}$  are SISO LTI and trial-invariant. As the system  $\mathbf{G}$ , a discrete-time rational transfer function is described as

$$\mathbf{G} = \mathbf{A}^{-1}(z)\mathbf{B}(z) \quad (1)$$

$$\mathbf{A}(z) = \sum_{k=0}^n a_k z^{-k}, \quad \mathbf{B}(z) = \sum_{k=1}^n b_k z^{-k} \quad (2)$$

where  $n$  denotes the order of the system and  $a_k, b_k \in \mathbb{R}$ .

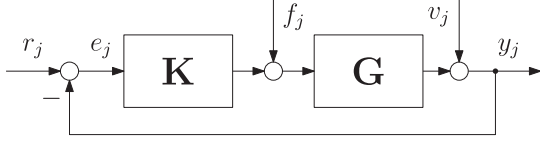


Fig. 2. Block diagram of closed-loop system.  $j$  denotes the iteration number of the motion task.

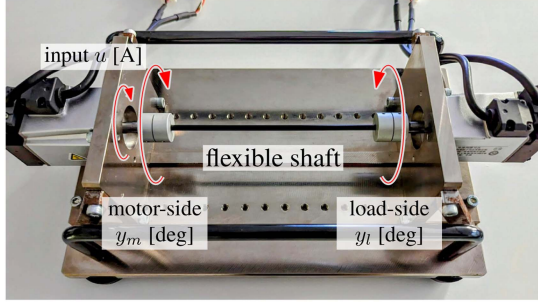


Fig. 3. Experimental setup of two-mass motion system  $G$ .

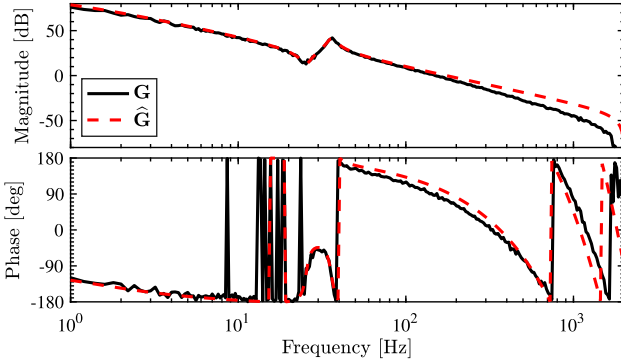


Fig. 4. Frequency response data  $G$  and model  $\hat{G}$  of the system.

For this study, as a general system with rational transfer functions, controlling the motor-side position of the two-mass motion system presented in Fig. 3 is considered. With dynamics associated with the flexible shaft, antiresonance and resonance appear as in Fig. 4. As a result, the system becomes rational, where the poles are associated with the rigid-body dynamics and resonance, and zeros are associated with the antiresonance.

Subscript  $j$  denotes the trial number of execution, with  $r_j$  denoting the reference,  $f_j$  the FF signal,  $y_j$  the measured output,  $v_j$  the measurement noise, and  $e_j$  the measured error given by

$$e_j = S r_j - J f_j - S v_j \quad (3)$$

with  $S = (1 + G K)^{-1}$  and  $J = S G$  denoting the sensitivity and process sensitivity function, respectively. For simplicity, throughout this study,  $v_j = 0$  is assumed.

### B. Model-Based FF Design

In model-based FF control, the FF input is designed as

$$f_{j+1} = F(\theta) r_{j+1} \quad (4)$$

with a parameterized FF controller  $F(\theta)$  defined as follows.

**Definition 1 (Parameterized FF controller):** Given  $\theta$ , the parameterized FF controller  $F(\theta)$  is constructed by

$$F(\theta) = B^{-1}(\theta) A(\theta), \quad \theta = [\theta^A, \theta^B]^\top \in \mathbb{R}^{(n_a + n_b) \times 1} \quad (5a)$$

$$A(\theta) = \sum_{i=0}^{n_a-1} \Psi^A[i] \theta^A[i], \quad B(\theta) = 1 + \sum_{i=0}^{n_b-1} \Psi^B[i] \theta^B[i] \quad (5b)$$

with user-defined basis functions  $\Psi^A(z) \in \mathbb{R}^{n_a \times 1}$  and  $\Psi^B(z) \in \mathbb{R}^{n_b \times 1}$ .

From (3), the tracking error with the FF input (4) is

$$e_{j+1} = S(1 - G F(\theta)) r_{j+1} \quad (6)$$

motivating the design of  $F(\theta) = G^{-1}$ . For standard model-based FF, the FF parameter  $\theta$  is tuned manually in a trial-and-error fashion, requiring additional design effort as the number of tuning parameters  $n_a + n_b$  increases [17].

**Remark 1:** In polynomial basis function design, only  $\Psi^A$  is used, i.e.,  $B(\theta) = 1$ , whereas in rational basis function design,  $\Psi^B$  is used to represent the poles of the inverse system  $G^{-1}$ , i.e., the zeros of  $G$ . This fundamentally increases the expressivity of basis functions, leading to enhanced task-flexibility and tracking performance.

### C. ILC Frameworks

Several design frameworks exist for ILC. In this study, F-ILC and B-ILC are considered and presented ahead.

**1) Frequency-Domain ILC (F-ILC):** The aim of F-ILC is to determine the FF  $f_{j+1}$ , achieving perfect tracking control for trial-invariant reference, i.e.,  $r_{j+1} = r_j$ . In F-ILC, this is achieved by designing  $f_{j+1}$  based on

$$f_{j+1} = Q(f_j + L e_j) \quad (7)$$

with learning filter  $L$  and robustness filter  $Q$  designed by the user.

From (3) and (7), the propagation of error and FF input are expressed as

$$e_{j+1} = Q(1 - L J) e_j + (1 - Q) S r_j \quad (8a)$$

$$f_{j+1} = Q(1 - L J) f_j + Q L S r_j \quad (8b)$$

with an assumption of  $r_{j+1} = r_j$ . From (8), it is guaranteed that both  $\|e\|_2$  and  $\|f\|_2$  converge monotonically within the entire frequency domain  $\omega$  when

$$|Q(e^{i\omega})(1 - L(e^{i\omega})J(e^{i\omega}))| < 1 \quad \forall \omega \in [0, \pi] \quad (9)$$

is satisfied [8]. This leads to the asymptotic error  $\lim_{j \rightarrow \infty} e_j = e_\infty$  of F-ILC

$$e_\infty = \frac{(1 - Q)S}{1 - Q(1 - L J)} r_\infty. \quad (10)$$

When (9) is satisfied with  $Q = 1$ , perfect tracking  $e_\infty = 0$  is achieved in a noiseless environment, i.e.,  $v_j = 0$ . From this fact and the existence of monotonic convergence condition (9), requirements R2 and R3 are satisfied. However, due to the requirement of  $r_{j+1} = r_j$  for (10), requirement R1 is violated.

2) **Basis Function ILC (B-ILC)**: The aim of B-ILC is to determine the FF  $f_{j+1}$ , achieving high tracking-performance even for tasks with trial-varying references, i.e.,  $r_{j+1} \neq r_j$ . In B-ILC, this is achieved by learning the optimal FF parameter  $\theta^*$  for (4), which is determined iteratively by minimizing criterion  $V(\theta_{j+1})$  defined in Definition 2.

**Definition 2 (Criterion for B-ILC)**: The performance criterion for B-ILC is given by

$$V(\theta_{j+1}) = \frac{1}{2} \|\hat{e}_{j+1}\|^2, \quad \theta_{j+1} = \arg \min_{\theta_{j+1}} V(\theta_{j+1}) \quad (11)$$

where

$$\hat{e}_{j+1} = e_j + \hat{J}(\underline{f}_j - \underline{f}_{j+1}). \quad (12)$$

**Remark 2**: Weighted sums of the FF input  $\|\underline{f}_{j+1}\|^2$  and FF input update  $\|\underline{f}_{j+1} - \underline{f}_j\|^2$  can be added to the criterion for enforcing robust learning (R3) [18]. To facilitate the presentation, the basic form (11) is used.

Given criterion (11), the optimal FF parameter update law is formulated as

$$\theta_{j+1} = Q_j \theta_j + L_j e_j \quad (13)$$

where the optimal  $Q_j$  and  $L_j$  are obtained through a sequential updating scheme as in [10] and [14].

In B-ILC, due to the FF input being composed based on the subsequent reference  $r_{j+1}$ , task flexibility against  $r_{j+1} \neq r_j$  is achieved satisfying requirement R1. While FF parameter optimization leads to enhanced tracking-performance compared to model-based FF, system  $G$  often contain dynamics unmodeled by (5). This leads to  $e_\infty \neq 0$  typically larger than (10) of F-ILC, partially violating requirement R2.

**Remark 3**: One of the specific challenges in the rational basis function framework is an optimization of  $\theta_{j+1}$ . For polynomial basis functions, i.e.,  $F(\theta_{j+1})\underline{r} = \Psi^A(r)\theta_{j+1}$ , criterion (11) is quadratic in  $\theta_{j+1}$ , providing an analytical solution for update (13). However, with a rational FF structure ( $B(\theta) \neq 1$ ), criterion (11) typically becomes nonconvex. To address this issue, several numerical solutions have been proposed such as using the Steiglitz-McBride approach [13] and instrumental variable approach [10], [14]. In this study, a sequential update law based on [14] is utilized.

## D. Problem Description

The problem addressed in this study is to develop an ILC framework satisfying both high task-flexibility (R1) and high tracking-performance (R2) with robust learning (R3). In B-ILC (rat), requirement R2 is partially violated typically due to the basis functions not including unmodeled dynamics, such as static friction. However, when the nonlinear dynamics are reproducible, i.e., trial-invariant, it can be learned by F-ILC.

In this study, the aim is to develop an ILC framework where 1) the rational basis function FF input  $f_{j+1}^B$  ensuring high task-flexibility, and 2) the frequency-domain FF input  $f_{j+1}^F$  only learning the residual dynamics, are simultaneously learned and

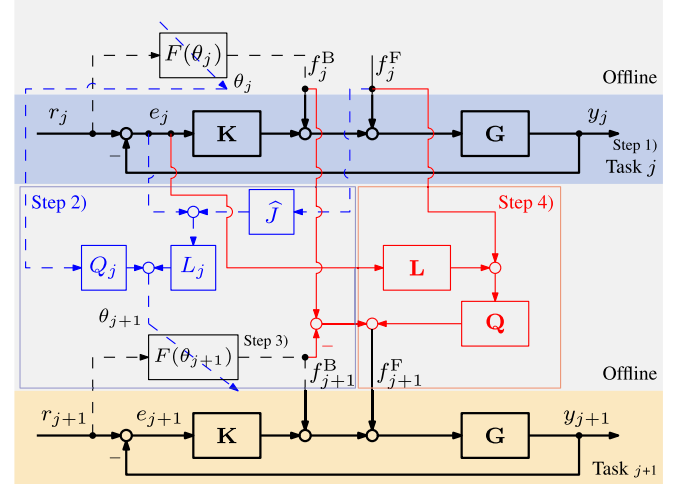


Fig. 5. Updating procedure of C-ILC. The flow of the time-domain update for  $f_{j+1}^B$  (→) and frequency-domain update for  $f_{j+1}^F$  (→) are illustrated.

combined as

$$f_{j+1} = f_{j+1}^B + f_{j+1}^F = F(\theta_{j+1})r_{j+1} + f_{j+1}^F \quad (14)$$

to satisfy all requirements R1–R3.

## III. INTEGRATING RATIONAL BASIS FUNCTIONS WITH F-ILC

In this section, the C-ILC (rat) framework is developed. The framework consists of the learning of basis function FF input  $f_{j+1}^B$  and frequency-domain FF input  $f_{j+1}^F$ , where the entire scheme is presented in Procedure 1 and illustrated in Fig. 5.

In addition, the robustness and potential of C-ILC to exceed the tracking-performance of F-ILC are presented. At last, stable inversion is addressed for realizing bounded solutions for possible scenarios when unstable filtering is encountered.

### A. Learning of Basis Function FF Input $f_{j+1}^B$

In basis function design of C-ILC, minimizing the criterion defined as Definition 3 is proposed.

**Definition 3 (Criterion for C-ILC  $f_{j+1}^B$ )**: The performance criterion for C-ILC is given by

$$V(\theta_{j+1}) = \frac{1}{2} \|\hat{e}_{j+1}^\theta\|^2, \quad \theta_{j+1} = \arg \min_{\theta_{j+1}} V(\theta_{j+1}) \quad (15)$$

where

$$\hat{e}_{j+1}^\theta = \hat{e}_{j+1} + \hat{J} \underline{f}_{j+1}^F \quad (16)$$

is the predicted virtual error solely induced by  $\underline{f}_{j+1}^F$ .

**Remark 4**: The aim of using (15) instead of (11) is to eliminate the (typically bad) influence of frequency-domain FF input  $f_{j+1}^F$  for FF parameter optimization [16]. When  $f = f^B + f^F$ , due to the existence of  $f^F$ , regularization of error  $e$  in (3) does not necessarily result in  $f^B \approx G^{-1}r$  as in (6). Similar approaches have been proposed to suppress multiperiod disturbance in repetitive control [19].

**Algorithm 1:** (Update of Developed Rational C-ILC).

Design

- Convolution matrix  $\hat{J}$  of process sensitivity model  $\hat{J}$
  - Initial  $f_0^B$  based on initial FF parameter  $\theta_0$
  - Initial  $f_0^F$
  - Learning filter  $L$  for frequency-domain learning
  - Robustness filter  $Q$  for frequency-domain learning and start with  $j = 0$ .
- 1) Perform the  $j$ th experiment and measure  $e_j$ .
  - 2) Follow the iterative steps to determine  $\theta_{j+1}$ .
    - a) Set  $\theta_{j+1}^{<0>} = \theta_j$  and initialize  $k = 0$ .
    - b) Construct  $L_j^{<k>}$  and  $Q_j^{<k>}$  based on  $\theta_{j+1}^{<k>}$  and (24).
    - c) Calculate  $\theta_{j+1}^{<k+1>} = Q_j^{<k>} \theta_j + L_j^{<k>} (e_j + \hat{J} f_j^F)$ .
    - d) Until a stopping condition is met [10], increment  $k \rightarrow k + 1$  and return to 2b). Else, set  $\theta_{j+1}^{<k+1>} = \theta_{j+1}^{<k>}$ .
  - 3) Construct  $F(\theta_{j+1})$  based on (5) and form  $f_{j+1}^B$  based on the reference  $r_{j+1}$ . When  $F(\theta_{j+1})$  has unstable poles, create a bounded  $f_{j+1}^B$  using stable inversion.
  - 4) Construct  $f_{j+1}^F$  based on (26).
    - a) Reset  $f_{j+1}^F = 0$  when  $r_{j+1} \neq r_j$ .
    - 5) Increment  $j \rightarrow j + 1$  and return to 1).

For optimizing (15), Lemmas 1 and 2 are derived for introducing  $\hat{e}_{j+1}^\theta$  in (16) to C-ILC (rat).

**Lemma 1** (Predicted virtual error of C-ILC): The predicted virtual error of C-ILC is derived as

$$\hat{e}_{j+1}^\theta = B^{-1}(\theta_{j+1}) \tilde{e}_j - \Phi(\theta_{j+1}) \theta_{j+1} \quad (17)$$

where

$$\tilde{e}_j = e_j + \hat{J} f_j \quad (18)$$

$$\Phi(\theta_{j+1}) = B^{-1}(\theta_{j+1}) \left[ \hat{J} \Psi^A(r_j), -\Psi^B(e_j + \hat{J} f_j) \right] \quad (19)$$

$$\Psi^A(r_j) = [\Psi^A[0]r_j, \Psi^A[1]r_j, \dots, \Psi^A[n_a - 1]r_j] \quad (20)$$

with  $f_j = f_j^F + f_j^B$ .

**Proof:** Follow from substituting  $e_j$  in (3) into  $\hat{e}_{j+1}^\theta$  in (16) with the use of (5) and (12). ■

**Lemma 2** (Optimal parameter  $\theta_{j+1}$  for C-ILC): The optimal  $\theta_{j+1}$  for (15) is derived as

$$\theta_{j+1} = \left( \left( \frac{\partial \hat{e}_{j+1}^\theta}{\partial \theta_{j+1}} \right)^\top \Phi(\theta_{j+1}) \right)^{-1} \left( \frac{\partial \hat{e}_{j+1}^\theta}{\partial \theta_{j+1}} \right)^\top B^{-1}(\theta_{j+1}) \tilde{e}_j \quad (21)$$

where

$$\frac{\partial \hat{e}_{j+1}^\theta}{\partial \theta_{j+1}} = B^{-1}(\theta_{j+1}) \left[ -\hat{J} \Psi^A(r_j), F(\theta_{j+1}) \hat{J} \Psi^B(r_j) \right]. \quad (22)$$

**Proof:** The proof follows from standard norm-optimal ILC [18] with substitution of (5) for the optimal condition  $\frac{\partial V(\theta_{j+1})}{\partial \theta_{j+1}} = 0$  [10], [14]. ■

From the aforementioned definitions and lemmas, the optimal parameter update for C-ILC is derived as follows.

**Theorem 1** (Optimal parameter update for C-ILC): Optimal parameter update for C-ILC is formulated as

$$\theta_{j+1} = Q_j \theta_j + L_j (e_j + \hat{J} f_j^F) \quad (23)$$

where

$$L_j = R_j B^{-1}(\theta_j) \quad (24a)$$

$$Q_j = L_j \left[ \hat{J} \Psi^A(r_j), \Psi^B(e_j + \hat{J} f_j^F) \right]$$

$$R_j = \left( \left( \frac{\partial \hat{e}_{j+1}^\theta}{\partial \theta_{j+1}} \right)^\top \Phi(\theta_{j+1}) \right)^{-1} \left( \frac{\partial \hat{e}_{j+1}^\theta}{\partial \theta_{j+1}} \right)^\top B^{-1}(\theta_{j+1}). \quad (24b)$$

**Proof:** Similar to (17),  $\tilde{e}_j$  in (18) can be rearranged as

$$\tilde{e}_j = B^{-1}(\theta_j) \left( e_j + \hat{J} f_j^F + \left[ \hat{J} \Psi^A(r_j), \Psi^B(e_j + \hat{J} f_j^F) \right] \theta_j \right). \quad (25)$$

Therefore, by substituting (25) into (21), (23) is derived. ■

Using the previous update, the basis function FF input  $f_{j+1}^B$  of C-ILC (rat) avoids interference from the parallelly learned frequency-domain FF input  $f_{j+1}^F$ , which is outlined next.

## B. Learning of Frequency-Domain FF Input $f_{j+1}^F$

In the frequency-domain design of C-ILC, the update of  $f_{j+1}^F$  is proposed as Definition 4.

**Definition 4** (Update of C-ILC  $f_{j+1}^F$ ): Update of  $f_{j+1}^F$  is given by

$$f_{j+1}^F = Q(f_j^F + L e_j) + f_j^B - f_{j+1}^B. \quad (26)$$

The aim of using (26) instead of (7) is for  $f_{j+1}^F$  to only learn the residual dynamics uncaptured by the update of  $f_{j+1}^B$ . By adding the term  $f_j^B - f_{j+1}^B$  to the standard update (7),  $f_{j+1}^F$  takes into account the dynamics already learned by  $f_{j+1}^B$ .

**Remark 5:** Note that update (26) involves the term  $f_{j+1}^B$ , i.e., the basis function FF input for the next iteration. However, this issue can be solved by calculating  $f_{j+1}^B$  beforehand, as demonstrated in Procedure 1. As presented in Theorem 1, derivation of  $f_{j+1}^B$  does not require  $f_{j+1}^F$ , enabling the proposed procedure.

## C. Robustness of C-ILC

Using (23) and the finite time-domain representation of (26)

$$f_{j+1}^F = Q^F(f_j^F + L^F e_j) + f_j^B - f_{j+1}^B \quad (27)$$

where  $Q^F \in \mathbb{R}^{N \times N}$  and  $L^F \in \mathbb{R}^{N \times N}$ , respectively, denote the finite-time convolution matrix corresponding to  $Q$  and  $L$ , the following update holds:

$$\begin{bmatrix} I_N & \Psi^A(r_{j+1}) \\ 0 & I_{n_a} \end{bmatrix} \begin{bmatrix} f_{j+1}^F \\ \theta_{j+1} \end{bmatrix}$$

$$= \begin{bmatrix} Q^F & \Psi^A(r_j) \\ L_j \hat{J} & Q_j \end{bmatrix} \begin{bmatrix} f_j^F \\ \theta_j \end{bmatrix} + \begin{bmatrix} Q^F L^F \\ L_j \end{bmatrix} e_j. \quad (28)$$

For simplicity,  $f_{j+1}^B = \Psi^A(r_{j+1})\theta_{j+1}$  is assumed. By substituting (3) and (14) into (29), the update can be written as

$$\begin{bmatrix} f_{j+1}^F \\ \theta_{j+1} \end{bmatrix} = \underbrace{\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}}_H \begin{bmatrix} f_j^F \\ \theta_j \end{bmatrix} + \begin{bmatrix} Q^F L^F - \Psi^A(r_{j+1})L_j \\ L_j \end{bmatrix} S_{r_j} \quad (29)$$

where

$$H_{11} = Q^F(I_N - L^F J) - \Psi^A(r_{j+1})L_j(\hat{J} - J) \quad (30)$$

$$H_{12} = (I_N - Q^F L^F J)\Psi^A(r_j) - \Psi^A(r_{j+1})(Q_j - L_j J\Psi^A(r_j)) \quad (31)$$

$$H_{21} = L_j(\hat{J} - J) \quad (32)$$

$$H_{22} = Q_j - L_j J\Psi^A(r_j). \quad (33)$$

Therefore, update (29) converges when

$$\rho(H) < 1 \quad (34)$$

where  $\rho(\cdot)$  denotes the spectral radius, i.e.,  $\rho(\cdot) = \max_i |\lambda_i(\cdot)|$ .

*Remark 6:*  $\rho(H) \approx \rho(H_{11}) \approx \rho(Q^F(I_N - L^F J))$  holds for typical setups. Using the maximum singular value denoted by  $\bar{\sigma}(\cdot)$ , (9) implies  $\bar{\sigma}(Q^F(I_N - L^F J)) < 1$  [8], [15]. From the well-known fact that  $\rho(\cdot) \leq \bar{\sigma}(\cdot)$ , the convergence condition of C-ILC can be approximately measured using (9).

#### D. Exceeding Tracking Performance of F-ILC

Assuming convergence of  $f_{j+1}^F$  and  $f_{j+1}^B$  provided from the previous section, the following lemma is derived for the asymptotic error of C-ILC.

*Lemma 3 (Asymptotic error of C-ILC):* Under assumption of monotonic convergence of  $f_{j+1}^F$  and  $f_{j+1}^B$ , the asymptotic error of C-ILC is

$$e_\infty = \frac{(1 - \mathbf{Q})\mathbf{S}(1 - \mathbf{G}\mathbf{F}(\theta_\infty))}{1 - \mathbf{Q}(1 - \mathbf{L}\mathbf{J})} r_\infty \quad (35)$$

where  $\lim_{j \rightarrow \infty} \mathbf{F}(\theta_j) = \mathbf{F}(\theta_\infty)$ .

*Proof:* Similar to the derivation of (10) for F-ILC, follows from using (3), (14), and (26), deriving the propagation of error and FF input for C-ILC [16]. ■

An interesting observation is that comparing (10) and (35), the asymptotic error of C-ILC is multiplied by  $(1 - \mathbf{G}\mathbf{F}(\theta_\infty))$  than that of F-ILC. Therefore, for frequency  $\omega$  where

$$|1 - \mathbf{G}(e^{i\omega})\mathbf{F}(\theta_\infty, e^{i\omega})| < 1 \quad (36)$$

C-ILC exceeds the tracking performance of F-ILC, depending on the quality of the learned  $\mathbf{F}(\theta_\infty)$ .

While F-ILC can only learn the system dynamics below the bandwidth of robustness filter  $\mathbf{Q}$ , B-ILC learns the parameters, which fit best to the structure of basis functions. Typically, this leads to B-ILC achieving  $\mathbf{F}(\theta_\infty) \approx \mathbf{G}^{-1}$  not limited by the bandwidth of  $\mathbf{Q}$ . Thus, C-ILC has the potential to learn

the dynamics over the bandwidth of  $\mathbf{Q}$ , exceeding the tracking performance against repetitive tasks compared to F-ILC.

#### E. Stable Inversion

Constructing  $Q_j$  and  $L_j$  in (23) or  $f_{j+1}^B$  in (14) can potentially involve filtering of unstable  $\mathbf{B}(\theta_{j+1})^{-1}$  and  $\mathbf{F}(\theta_{j+1})$ . To deal with possible instability, stable inversion is utilized.

Stable inversion [20] is a noncausal inversion technique, which yields an exact system inversion with infinite time preview. In situations where filtering can be done offline, e.g., ILC, noncausal filtering can be conducted. See the work in [20] and [21] for further details.

### IV. EXPERIMENTAL VALIDATION

In this section, the developed C-ILC (rat) framework is validated with a two-mass motion system. To investigate the effectiveness of the developed framework, the results are compared with pre-existing ILC frameworks, i.e., F-ILC, B-ILC (pol), B-ILC (rat), and C-ILC (pol), using unified conditions stated in Sections IV-A and IV-B.

#### A. Experimental Setup

In this study, the two-mass motion system shown in Fig. 3 is used as a benchmark system for a high-precision positioning stage. The system is controlled at a sampling time of  $T_s = 0.25$  ms with a stabilizing FB controller

$$\mathbf{K} = \frac{0.21028(1 - 0.9875z^{-1})}{1 - 0.6718z^{-1}}. \quad (37)$$

For the system model

$$\begin{aligned} \hat{\mathbf{G}} &= \frac{3.6712 \times 10^{-2}(1 + 0.9987z^{-1})}{(1 - 0.9978z^{-1})(1 - z^{-1})} \\ &\times \frac{1 - 1.994z^{-1} + 0.9961z^{-2}}{1 - 1.991z^{-1} + 0.9944z^{-2}} z^{-6} \end{aligned} \quad (38)$$

is designed with the bode plot illustrated as Fig. 4.

#### B. Learning Setup

**1) Learning Task:** The objective of this validation is to test both, tracking performance against repetitive tasks ( $r_{j+1} = r_j$ ) and task flexibility against nonrepetitive tasks ( $r_{j+1} \neq r_j$ ). To validate this, two reference trajectories shown in Fig. 6 are utilized. The blue solid line is used as the reference for the first 15 trials while the red dashed line is used as the reference for the last 15 trials.

**2) Frequency-Domain Design:** For the learning filter used for F-ILC and C-ILC,  $\mathbf{L} = \hat{\mathbf{J}}^{-1} = \hat{\mathbf{G}}^{-1}(1 + \hat{\mathbf{G}}\mathbf{K})$  is designed. Using the  $\mathbf{J}$  obtained from Fig. 4, a tenth-order zero-phase filter with a bandwidth of 120 Hz is designed for robustness filter  $\mathbf{Q}$ , satisfying monotonic convergence condition (9). The frequency-domain FF input is initialized as  $f_0^F = 0$  and reset to 0 whenever the task changes.

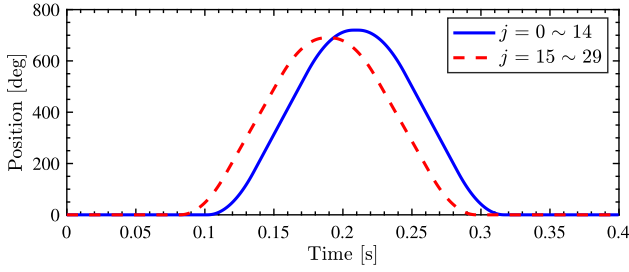


Fig. 6. Trajectory  $r_j$ . For validating both, high tracking-performance against repetitive tasks ( $r_{j+1} = r_j$ ) and high task-flexibility against non-repetitive tasks ( $r_{j+1} \neq r_j$ ), (—) and (---) are performed for the first and last 15 trials, respectively.

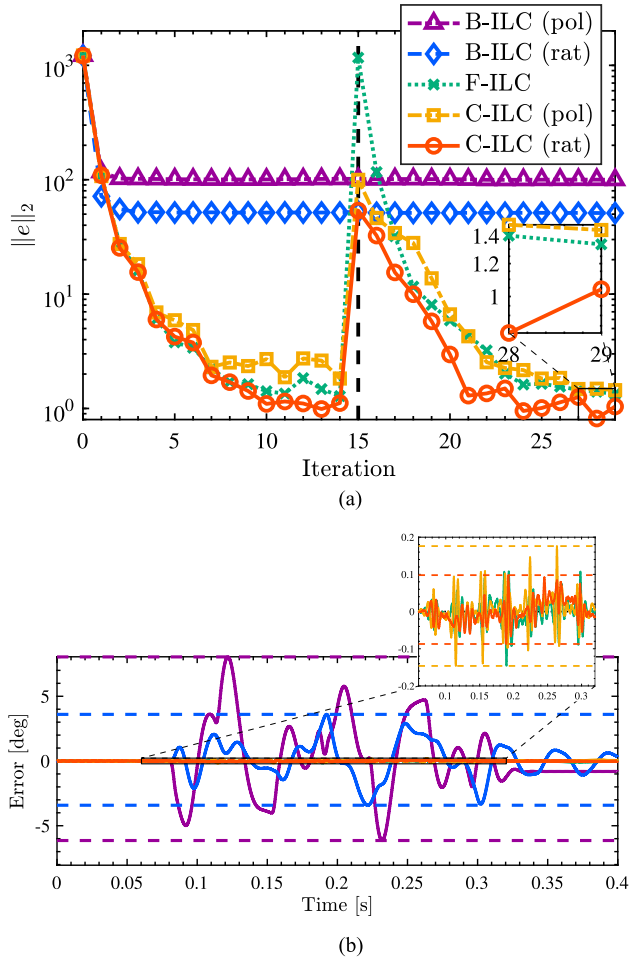


Fig. 7. Developed C-ILC (rat) (—○—) achieves as high task-flexibility as B-ILC (rat) (—◇—), higher than C-ILC (pol) (—□—) when task changes at  $j = 15$ . In terms of tracking performance, F-ILC (—×—) and both C-ILC frameworks (—, —) achieve high performance far exceeding both B-ILC frameworks (—, —). (a) Error norm per iteration. (b) Error signal of final iteration.

### 3) Basis Function Design:

$$\Psi^A = \left[ \frac{1-z^{-1}}{T_s}, \left( \frac{1-z^{-1}}{T_s} \right)^2, \left( \frac{1-z^{-1}}{T_s} \right)^3, \left( \frac{1-z^{-1}}{T_s} \right)^4 \right]$$

$$\Psi^B = \left[ \frac{1-z^{-1}}{T_s}, \left( \frac{1-z^{-1}}{T_s} \right)^2 \right]. \quad (39)$$

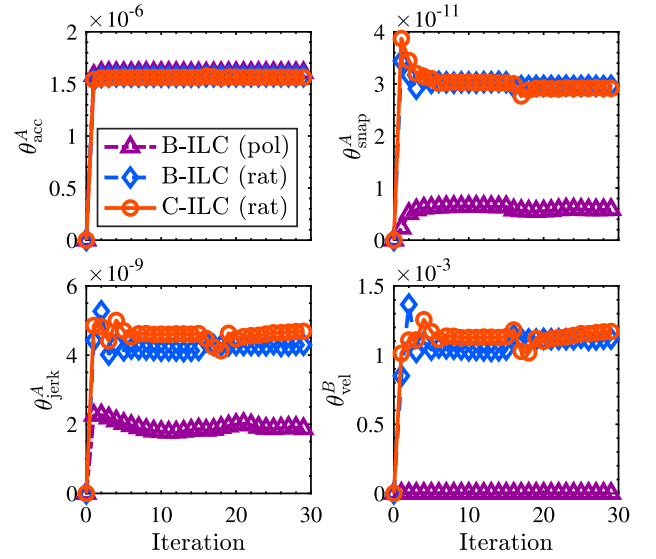


Fig. 8. FF parameter  $\theta_{acc}^A$ ,  $\theta_{jerk}^A$ ,  $\theta_{snap}^A$ , and  $\theta_{vel}^B$  of  $\theta_j$  learned by the basis functions. Using rational basis functions (—◇—), the differences between polynomial basis functions (—△—) are especially observed for the jerk and snap, which are associated with the high-order flexible dynamics. Moreover, it is observed that C-ILC (rat) (—○—) is able to achieve identical learning as that of B-ILC (rat) (—◇—), resulting to similar task-flexibility displayed in Fig. 7(a).

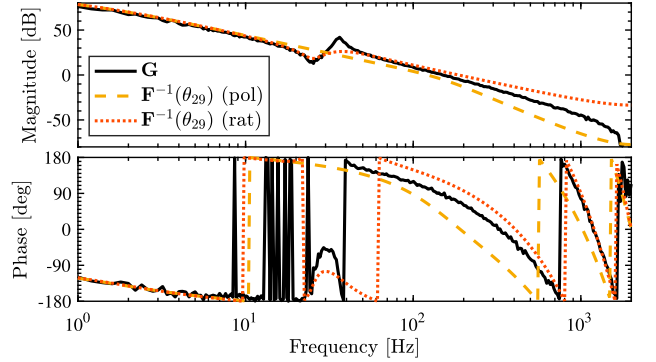


Fig. 9. Frequency response data of system  $G$  (—) and learned FF controller  $F(\theta)$  of C-ILC (pol) (---) and C-ILC (rat) (····). The antiresonance of the system is successfully captured by using rational basis functions, resulting in enhanced task-flexibility and tracking performance in Fig. 7.

The basis functions used for polynomial basis function design and rational basis function design are, respectively, denoted as  $\Psi = [\Psi^A]$  and  $\Psi = [\Psi^A, \Psi^B]$ . FF parameters corresponding to the basis functions in (39) are defined as  $\theta = [\theta_{vel}^A, \theta_{acc}^A, \theta_{jerk}^A, \theta_{snap}^A, \theta_{vel}^B, \theta_{acc}^B]$  and initialized as  $\theta_0 = [10^{-14}, 0, 0, 0, 0, 0]$ . In this validation, the relative degree of  $G$  is considered, i.e.,  $F(\theta_j, z) = z^6 B^{-1}(\theta_j, z) A(\theta_j, z)$ , and stable inversion is utilized when  $F(\theta_j)$  becomes unstable.

## C. Experimental Results

1) *High Tracking-Performance of C-ILC*: Fig. 7(a) shows the error norms per trial in F-ILC, B-ILC, and C-ILC. It is demonstrated that while repetitive tasks are performed, assured by

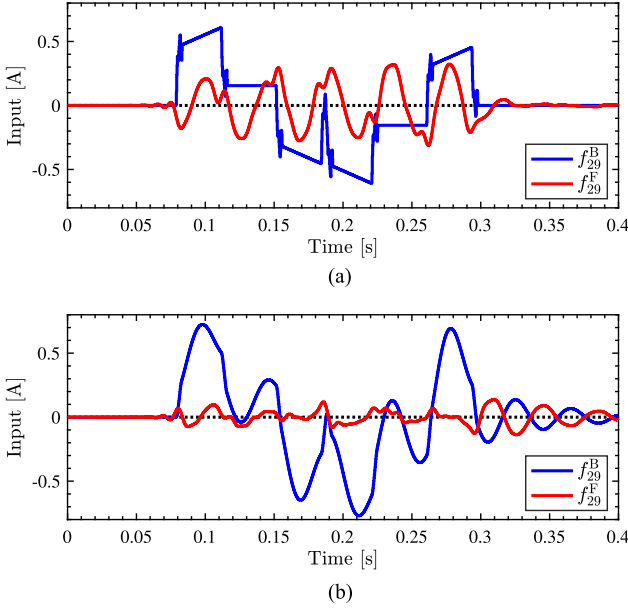


Fig. 10. Comparison of each learned FF input for the C-ILC framework. The input is mainly composed of the basis function component  $f_{29}^B$  (—) while frequency-domain component  $f_{29}^F$  (—) compensates the residual dynamics. (a) Learned FF input of C-ILC (pol). (b) Learned FF input of C-ILC (rat).

the designed robustness filter  $\mathbf{Q}$ , C-ILC and F-ILC successfully converge in about 10 trials, resulting in high tracking-performance far exceeding B-ILC. Fig. 7(b) shows the tracking error comparison of the final iteration. It is observed that while B-ILC exhibits an error signal in the low-frequency range, F-ILC and C-ILC achieve high tracking-performance by utilizing the frequency-domain learning targeted within the bandwidth of  $\mathbf{Q}$ . In addition, the best performing error norm  $\|e\|_2$  of C-ILC (rat) is reduced by 44% compared to that of C-ILC (pol). This is due to C-ILC (rat) learning a  $\mathbf{F}(\theta)$  more accurate to the system than C-ILC (pol), especially over the bandwidth of  $\mathbf{Q}$ , as mentioned in Section III-D.

**2) Task Flexibility of C-ILC:** The results of the learned FF parameters are shown in Fig. 8. First, the results of B-ILC (pol) and B-ILC (rat) show a significant difference in the jerk and snap, which are associated with the high-order flexible dynamics. Fig. 9 shows that the rational FF controller effectively compensates for the flexible dynamics by capturing the antiresonance of the system with rigid dynamics associated with the system poles. This enables B-ILC (rat) to have a better performance than B-ILC (pol) in Fig. 7(a). Second, the results of B-ILC (rat) and C-ILC (rat) show identical learning. This results in C-ILC (rat) exhibiting similar task-flexibility in the 15th trial of Fig. 7(a), exceeding that of C-ILC (pol) and B-ILC (pol).

**3) Learned FF Input of C-ILC:** Fig. 10 shows the results of the learned FF input  $f_{29} = f_{29}^F + f_{29}^B$  for C-ILC. For both C-ILC (pol) and C-ILC (rat),  $f_{29}^B$  consists of the main component of FF input while  $f_{29}^F$  compensates for the residual dynamics uncaptured by the basis functions. Owing to rational basis functions being richer than polynomial basis functions in expressivity, the magnitude of  $f_{29}^F$  is smaller for C-ILC (rat) while the motion is

active during 0.08–0.30 s. This is desirable in terms of task flexibility, as  $f_{29}^F$  is reset to zero whenever the task changes. After the motion ends, the undesirable effect of rational  $\mathbf{F}(\theta)$  magnifying the antiresonance frequency of the error is effectively canceled out by the learning of  $f_{29}^F$ .

## V. CONCLUSION

In this study, an ILC framework combining B-ILC (rat) and F-ILC is developed. By developing a parallel learning scheme avoiding interference of the distinct frameworks, high task-flexibility as that of B-ILC (rat) and high tracking-performance as that of F-ILC are simultaneously achieved. The framework is experimentally validated on a general motion system with dominant pole and zero dynamics. The results verify the developed C-ILC (rat) framework 1) learn FF parameters identical to that of B-ILC (rat) enabling high task-flexibility, while 2) utilize frequency-domain learning to enable high tracking-performance, compensating the residual dynamics overlooked by the basis functions. Additionally, by C-ILC (rat) utilizing rational basis functions, flexible dynamics associated with system zeros are successfully learned, which otherwise would not have been captured by only using polynomial basis functions. This suggests that C-ILC (rat) is effective for high-precision mechatronic systems, which often have parasitic dynamics associated with system zeros and perform both repetitive and nonrepetitive motion tasks.

Future research focuses on extending the C-ILC theory to more complicated systems, such as multivariable systems.

## REFERENCES

- [1] S. Mishra, J. Coaplen, and M. Tomizuka, "Precision positioning of wafer scanners segmented iterative learning control for nonrepetitive disturbances [applications of control]," *IEEE Control Syst.*, vol. 27, no. 4, pp. 20–25, Aug. 2007.
- [2] S. H. Van Der Meulen, R. L. Tousain, and O. H. Bosgra, "Fixed structure feedforward controller design exploiting iterative trials: Application to a wafer stage and a desktop printer," *J. Dyn. Syst., Meas. Control*, vol. 130, no. 5, pp. 1–16, Sep. 2008.
- [3] N. Nikooinajad, M. Maroufi, and S. O. R. Moheimani, "Iterative learning control for video-rate atomic force microscopy; iterative learning control for video-rate atomic force microscopy," *IEEE/ASME Trans. Mechatron.*, vol. 26, no. 4, pp. 2127–2138, Aug. 2021.
- [4] K. Saiki, A. Hara, K. Sakata, and H. Fujimoto, "A study on high-speed and high-precision tracking control of large-scale stage using perfect tracking control method based on multirate feedforward control," *IEEE Trans. Ind. Electron.*, vol. 57, no. 4, pp. 1393–1400, Apr. 2010.
- [5] M. Steinbuch, T. Oomen, and H. Vermeulen, "Motion control, mechatronics design, and Moore's law," *IEEE J. Ind. Appl.*, vol. 11, no. 2, pp. 245–255, 2022.
- [6] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronics systems," in *Proc. IEEE 23rd Conf. Decis. Control*, 1984, pp. 1064–1069.
- [7] K. L. Moore, *Iterative Learning Control for Deterministic Systems*. London, U.K.: Springer, 2012.
- [8] M. Norrlöf and S. Gunnarsson, "Time and frequency domain convergence properties in iterative learning control," *Int. J. Control*, vol. 75, no. 14, pp. 1114–1126, 2002.
- [9] A. Bristow, M. D. Tharayil, and A. G. Andrew, "A survey of iterative learning control," *IEEE Control Syst. Mag.*, vol. 26, no. 3, pp. 96–114, Jun. 2006.
- [10] L. Blanken, F. Boeren, D. Bruijnen, and T. Oomen, "Batch-to-batch rational feedforward control: From iterative learning to identification

approaches, with application to a wafer stage," *IEEE/ASME Trans. Mechatron.*, vol. 22, no. 2, pp. 826–837, Apr. 2017.

- [11] J. Van De Wijdeven and O. H. Bosgra, "Using basis functions in iterative learning control: Analysis and design theory," *Int. J. Control*, vol. 83, no. 4, pp. 661–675, 2010.
- [12] R. Pintelon and S. Johan, *System Identification: A Frequency Domain Approach*. Hoboken, NJ, USA: Wiley, 2012.
- [13] J. Bolder and T. Oomen, "Rational basis functions in iterative learning control - with experimental verification on a motion system," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 2, pp. 722–729, Mar. 2015.
- [14] 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.
- [15] 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.*, vol. 21, no. 6, pp. 2716–2727, Dec. 2016.
- [16] K. Tsurumoto, W. Ohnishi, and T. Koseki, "Task flexible and high performance ILC: Preliminary analysis of combining a basis function and frequency-domain approach," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 1907–1912, Jan. 2023.
- [17] T. Oomen, "Advanced motion control for precision mechatronics: Control, identification, and learning of complex systems," *IEEE J. Ind. Appl.*, vol. 7, no. 2, pp. 127–140, 2018.
- [18] S. Gunnarsson and M. Norrlöf, "On the design of ILC algorithms using optimization," *Automatica*, vol. 37, pp. 2011–2016, 2001.
- [19] L. Blanken, P. Bevers, S. Koekebakker, and T. Oomen, "Sequential multi-period repetitive control design with application to industrial wide-format printing," *IEEE/ASME Trans. Mechatron.*, vol. 25, no. 2, pp. 770–778, Apr. 2020.
- [20] S. Devasia, D. Chen, and B. Paden, "Nonlinear inversion-based output tracking," *IEEE Trans. Autom. Control*, vol. 41, no. 7, pp. 930–942, Jul. 1996.
- [21] J. Van Zundert and T. Oomen, "On inversion-based approaches for feed-forward and ILC," *Mechatron.*, vol. 50, pp. 282–291, 2018.



**Kentaro Tsurumoto** received the M.Sc. degree in electrical engineering from the Department of Electrical Engineering and Information Systems, The University of Tokyo, Tokyo, Japan, in 2024. He is currently working toward the Ph.D. degree in electrical engineering with the Department of Electrical Engineering and Information Systems, The University of Tokyo.

His research interests include high-precision motion control and learning control.

Mr. Tsurumoto is a Student Member of The Institute of Electrical Engineers of Japan.



**Wataru Ohnishi** received the B.E., M.Sc., and Ph.D. degrees in electrical engineering from the Department of Electrical Engineering and Information Systems, The University of Tokyo, Tokyo, Japan, in 2013, 2015, and 2018, respectively.

He is currently an Associate Professor with the Department of Electrical Engineering and Information Systems, Graduate School of Engineering, The University of Tokyo. He held a visiting position with the Eindhoven University of Technology. His research interests include

high-precision motion control and optimization.

Dr. Ohnishi is a Senior Member of The Institute of Electrical Engineers of Japan.



**Takafumi Koseki** received the Ph.D. degree in electrical engineering from the Department of Electrical Engineering and Information Systems, The University of Tokyo, Tokyo, Japan, in 1992.

He is currently a Professor with the Department of Electrical Engineering and Information Systems, Graduate School of Engineering, The University of Tokyo. His research interests include public transport systems, particularly linear drives, and the analysis and control of traction systems.

tion systems.

Dr. Koseki is a Fellow Member of The Institute of Electrical Engineers of Japan.



**Max van Haren** received the M.Sc. degree (*cum laude*) in mechanical engineering in 2021 from the Eindhoven University of Technology, Eindhoven, The Netherlands, where he is currently working toward the Ph.D. degree in mechanical engineering with the Control Systems Technology group, Department of Mechanical Engineering, Eindhoven University of Technology.

His research interests include control and identification of sampled-data, multirate, and linear parameter-varying systems for mecha-

tronics.



**Tom Oomen** received the M.Sc. (*cum laude*) and the Ph.D. degrees in mechanical engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 2005 and 2010, respectively.

He is currently a Professor with the Department of Mechanical Engineering, Eindhoven University of Technology. He is also a part-time Full Professor with the Delft University of Technology, Delft, The Netherlands. He held visiting positions with KTH, Stockholm, Sweden, and

with The University of Newcastle, Australia. His research interests include data-driven modeling, learning, and control, with applications in precision mechatronics.

Dr. Oomen is a recipient of the 7th Grand Nagamori Award, the Corus Young Talent Graduation Award, IFAC 2019 TC 4.2 Mechatronics Young Research Award, 2015 IEEE Transactions on Control Systems Technology Outstanding Paper Award, 2017 IFAC Mechatronics Best Paper Award, 2019 IEEE Journal of Industry Applications Best Paper Award, and Veni and Vidi personal grant. He is currently a Senior Editor of the IEEE CONTROL SYSTEMS LETTERS (L-CSS) and Co-Editor-in-Chief of *IFAC Mechatronics*. He was an Associate Editor for *IFAC Mechatronics*, IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, and L-CSS. He is a member of the Eindhoven Young Academy of Engineering.