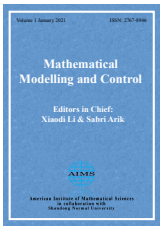
Mathematical Modelling and Control

https://www.aimspress.com/journal/mmc



MMC, 5(3): 321–337.
DOI: 10.3934/mmc.2025022
Received: 12 July 2025
Revised: 09 August 2025
Accepted: 18 August 2025
Published: 25 September 2025

Research article

Iterative learning control optimization strategy for feedback control systems with varying tasks

Fangmei Chen[1], **Hongfeng Tao**[1,*], **Zhihe Zhuang**[1], **Wojciech Paszke**[2] **and Vladimir Stojanovic**[3,*]

[1] Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi 214122, China

[2] Institute of Automation, Electronic and Electrical Engineering, University of Zielona Góra, Zielona Góra, 65-417, Poland

[3] Faculty of Mechanical and Civil Engineering, Department of Automatic Control, Robotics and Fluid Technique, University of Kragujevac, Kraljevo 36000, Serbia

* **Correspondence:** Email: taohongfeng@jiangnan.edu.cn, vladostojanovic@mts.rs.

Abstract: Iterative learning control (ILC) combined with feedback control is a common approach to repetitive systems with external disturbances, as it enables high tracking performance and guarantees time-domain stability. However, the variation of the reference trajectory in practical repetitive operations often degrades the control performance. To this end, this paper develops a feedback-based ILC to transfer the experience of repetitively operating a certain task to a brand new task without restriction on its time duration. This two-dimensional (2-D) design employs a parallel structure, where the ILC and the feedback controller are designed separately to achieve performance optimization. Then, the feedback plus feedforward controller is integrated into a new feedback controller with learning-based parameters. The convergence and robustness analysis of the design is given. Finally, numerical simulation experiments of a DC motor position control system verify the proposed scheme's effectiveness and robustness.

Keywords: iterative learning control; feedback control; parallel structure; performance optimization; varying task

1. Introduction

Iterative learning control (ILC) is an effective dynamic control algorithm for repetitive operations over a fixed time interval [1]. It incrementally updates the control input by learning from previous trials to improve tracking performance. In recent years, ILC has been extensively utilized in control areas such as batch processes [2], industrial robotic systems [3], precision motion control systems [4], and robot-assisted biomedical systems [5] due to its simple structure, virtually model-free, and learning properties. For more explanations of ILC, refer to the surveys [6–8] and references therein.

In practical industrial processes, the control performance of ILC is inevitably affected by factors such as modeling errors and model uncertainty. To address these challenges, researchers have made many efforts. A norm optimal ILC (NOILC) framework with time-varying weighting matrices was developed to improve control performance [9], while a data-driven adaptive ILC based on pointwise dynamic linearization was introduced for improved adaptability [10]. For linear systems with varying trial lengths and input constraints, an improved optimal ILC algorithm incorporating the primal-dual interior point method has been developed to address the input constraints [11]. However, these studies mainly focus on inter-trial learning while neglecting real-time feedback performance during the trial, which have certain limitations. Given that ILC is

essentially an open-loop control method [12], it lacks the ability to respond to real-time errors during execution. To overcome this limitation, integrating feedforward ILC with feedback control has emerged as an effective solution.

The integration of ILC and feedback combines short-term error suppression and long-term optimal learning, yielding better performance in dynamic and uncertain environments. To enhance system robustness, a two-stage control framework was proposed in [13], where feedback and ILC are decoupled by separately updating inputs for disturbance rejection and trajectory learning. A two-dimensional (2-D) linear quadratic Gaussian benchmark was introduced in [14] to evaluate the performance of ILC within the 2-D framework. To enhance the time-domain stability of ILC, a hybrid approach combining ILC and model predictive control (MPC) has been developed [15–17]. An integrated control strategy combining robust ILC with MPC was proposed in [16] to enable robust convergence of systems with parameter uncertainties. In [17], the integration of H_∞ control and MPC-ILC was introduced to mitigate tracking error fluctuations caused by reference changes. In addition, the roles of feedback control and ILC can also be separated [18]. An indirect adaptive ILC scheme via setpoint updating was proposed in [19]: The outer-loop ILC is used to improve the performance of the inner-loop feedback control. A state-feedback-based ILC method was proposed in [20], which allows the system to maintain good tracking performance under hard input constraints. Although these methods are effective in repetitive tasks, they are typically task-specific and may fail to maintain satisfactory control performance when reference trajectories or operating conditions change.

The assumption of strictly repetitive tasks frequently fails to hold in practical applications. In [21], an improved ILC scheme was proposed for systems with time lags and non-repeating reference signals, where the control signal was iteratively changed by modifying the reference signal. A robust indirect-type ILC method based on PID feedack was proposed in [22] to accommodate the uncertainty of time-varying processes by establishing sufficient constraints on linear matrix inequalities. In addition, an ILC scheme based on a generalized extended state observer was designed in [23] to overcome time-varying uncertainties and non-repeating perturbations. A novel rational feedforward adjustment method based on a two-degree-of-freedom (2-DOF) framework was proposed in [24], which achieves high-performance tracking of varying trajectories by directly mapping the feedforward signals learned from the dual-loop ILC to the corresponding reference points. Although the above methods can improve the system tracking performance under certain non-repetitive conditions, they tend to be inflexible when dealing with unknown varying tasks. It still needs to relearn the task to adapt to the new reference trajectory, which reduces the industrial efficiency to some extent.

Inspired by the above analysis, this paper develops a novel feedback-based ILC method which embeds ILC as a feedforward component into a closed-loop feedback system. It leverages the iterative learning capability of ILC to guide the feedback controller in adapting to changing task trajectories. The proposed method adopts a 2-D parallel control structure: PID feedback is introduced along the time axis for real-time disturbance rejection, while a NOILC strategy [25] performs inter-trial optimization along the iteration axis. The integration of these two components enhances the system's robustness against uncertainties and non-repetitive disturbances. Specifically, the proposed method combines the trial-to-trial learning capability of ILC with feedback control by repeatedly performing a nominal task to obtain the system's optimal control behavior. This learned behavior is then embedded into a feedback controller via least-squares fitting, enabling rapid adaptation to new task trajectories without re-learning. This method is particularly suitable for industrial scenarios similar to those described in [26] where the system needs to perform operations repeatedly, but the task trajectory changes frequently, e.g., robot assembly and motor motion control. A servomotor position control system is chosen as a numerical simulation example to verify the effectiveness.

The principal contributions of this paper can be summarized as follows:

- A new feedback-based ILC method is proposed in this paper, using a 2-D parallel control structure with the NOILC strategy for inter-trial learning and PID control for real-time disturbance suppression, which effectively copes with dynamic variations and non-

repetitive disturbances in the system.

- By integrating the feedback plus feedforward controller into a learning-based feedback controller, the historical learning experience can be transferred to a new task without limiting its duration. Efficient tracking of unknown changing trajectories can be achieved without repeated learning.
- The robust convergence of the feedback-based ILC system is theoretically derived and demonstrated, and the effectiveness of the proposed scheme in coping with unknown changing trajectories is verified through simulation.

The structure of this paper is organized as follows: The problem formulation is given in Section 2. Then, in Section 3, the design of the proposed two-stage algorithm is presented. Section 4 gives the performance analysis and convergence analysis of the proposed control scheme. The validity of the proposed algorithm is verified in Section 5. Finally, conclusions are given in Section 6.

Notation: $\mathbb{N}$ denotes the set of positive integers. $\mathbb{R}^n$ represents the set of n-dimensional real vectors. The superscript $\top$ represents the transposed component of a vector or a matrix. $\|\cdot\|$ denotes the Euclidean norm. The notation $\mathbf{0}$ is a zero vector or matrix with appropriate dimensions. $\|u\|_R^2 = u^\top Ru$ is the induced norm of a matrix defined in Hilbert space with weighting matrix R.

2. Preliminaries and problem formulation

This section commences with an exposition of the system dynamics in mathematical notation. Then, the framework of the feedback-based ILC system design is presented, along with an overview of the system's overall design objectives.

2.1. System description

In practical industrial motion control systems, a 2-DOF control architecture is commonly employed, as shown in Figure 1.

The control system consists of the controlled plant $H(z)$, a PID controller $C(\theta, z)$ with adjustable parameter θ, and an unknown feedforward controller $F(z)$. Here, y_d, u, and y represent the reference, input, and output signals, respectively, while f and η denote the feedforward input and external disturbance signals. Without considering the external disturbance η, the actual output y of the closed-loop system in Figure 1 can be expressed as follows:

$$y = \underbrace{\frac{H(z)C(\theta,z)}{1+H(z)C(\theta,z)}}_{\mathcal{T}(\theta,z)} y_d + \underbrace{\frac{1}{1+H(z)C(\theta,z)}}_{\mathcal{S}(\theta,z)} H(z)f, \tag{2.1}$$

where $\mathcal{T}(\theta, z)$ and $\mathcal{S}(\theta, z)$ are the complementary sensitivity and sensitivity functions, respectively. The system error is given by $e = \mathcal{S}y_d - \mathcal{S}Hf$, where its magnitude is affected by both the sensitivity function and the feedforward input. Enhancing overall system performance can thus be achieved by optimizing either the feedback controller or the feedforward controller. The transfer function $C(\theta, z)$ of the discrete PID controller can be obtained by the z-transform as

$$C(\theta, z) = K_p + K_i \frac{T_s z}{z-1} + K_d \frac{z-1}{T_s z}, \tag{2.2}$$

where T_s is the sampling period, and $\theta = [K_p, K_i, K_d]^\top$ represents the parameter vector. K_p, K_i, and K_d are the proportional, integral, and derivative gains, respectively, which play a crucial role in the PID control performance.

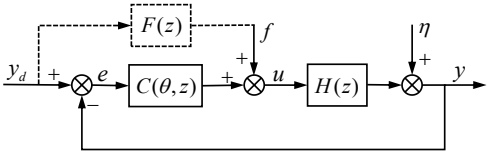


Figure 1. 2-DOF control system with stable feedback controller.

2.2. Feedback-based ILC system design

The fixed-parameter feedback control system cannot cope with the challenges of various uncertain disturbances in real industrial systems. For this reason, this paper proposes a two-stage control strategy, as illustrated in Figure 2.

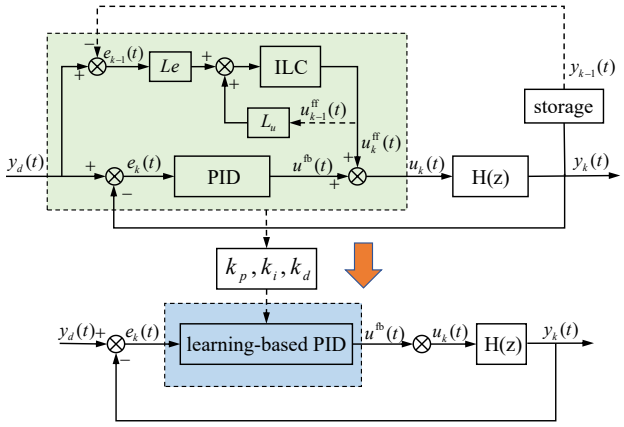


Figure 2. Parallel feed-forward feedback experience migration ILC block diagram.

Based on the feedback control structure, an ILC scheme is introduced to form a 2-D parallel framework, which combines real-time feedback along the time axis and historical learning along the iteration axis to continuously optimize system performance through repeated trials.

In a finite time interval $t \in [0, N]$, the control input to the system during the kth trial consists of two distinct components: the stabilizing feedback control and the feedforward ILC, which can be described as

$$u_k(t) = u_k^{\mathrm{ff}}(t) + u_k^{\mathrm{fb}}(t), \tag{2.3}$$

where $u_k^{\mathrm{ff}}(t) \in \mathbb{R}$ and $u_k^{\mathrm{fb}}(t) \in \mathbb{R}$ represent the ILC control input and PID controller output, respectively. As an open-loop feedforward control strategy, the ILC update law along the iterative axis is designed as

$$u_{k+1}^{\mathrm{ff}}(t) = P(u_k^{\mathrm{ff}}(t) + Le_k(t+1)), \tag{2.4}$$

where $e_k(t+1) = y_d(t+1) - y_k(t+1)$ represents the tracking error at time $t+1$ in the kth trial of the system. P and L are defined as the robustness filter and learning gain, respectively.

On the time axis, the PID is designed as

$$u_k^{\mathrm{fb}}(t) = K_p e_k(t) + K_i T_s \sum_{j=0}^{t} e_k(j) + \frac{K_d}{T_s}(e_k(t) - e_k(t-1)). \tag{2.5}$$

With the addition of ILC, the system output can be expressed as

$$y_k(t) = \underbrace{\frac{H(z)}{1 + H(z)C(\theta, z)}}_{G_c(\theta,z)} u_k^{\mathrm{ff}}(t) + \underbrace{\frac{H(z)C(\theta, z)}{1 + H(z)C(\theta, z)}}_{G_s(\theta,z)} y_d(t), \tag{2.6}$$

where $G_c(\theta, z)$ and $G_s(\theta, z)$ represent known linear discrete-time transfer operators.

Remark 2.1. *The combination of ILC and feedback control brings together the strengths of both approaches: ILC enables long-term learning through iterative optimization, while feedback control ensures real-time disturbance suppression and dynamic stability. Compared to pure ILC, the introduction of feedback improves robustness to modeling errors and external disturbances, and speeds up ILC convergence by compensating for initial errors. In addition, the use of feedback makes it easier to extend the feedback-based ILC framework to time-varying or nonlinear systems. In time-varying systems, feedback compensates for system parameter variations and disturbances in real time. For nonlinear systems [20], state feedback helps maintain bounded trajectories, supporting ILC algorithm design under local Lipschitz continuity.*

Remark 2.2. *In the feedback-based ILC structure, the control contributions from the ILC and PID feedback are easily separated. As the ILC converges, the feedback controller has less work to do.*

The system adopts time domain analysis under the lifting framework. To construct the lifted system representation, the corresponding impulse response coefficients of G_c and G_s are computed based on their unit impulse responses as follows:

$$G_c(h) = g_1 h^{-1} + g_2 h^{-2} + \cdots + g_N h^{-N}, \tag{2.7}$$

$$G_s(h) = r_1 h^{-1} + r_2 h^{-2} + \cdots + r_N h^{-N}, \tag{2.8}$$

where h is the forward time-shift operator, and the coefficients g_i and r_i are Markov parameters [6]. The infinite impulse responses are truncated at order N. Based on the truncated responses, the lifting matrices $G_d \in \mathbb{R}^{N\times N}$ and $G_r \in \mathbb{R}^{N\times N}$ are constructed as lower triangular Toeplitz matrices using the Markov parameters from (2.7) and (2.8), respectively. System (2.6) can be reconfigured into a lifting

framework within the iterative domain as follows:

$$y_k = G_d u_k^{\text{ff}} + \underbrace{G_r y_d}_{d}, \tag{2.9}$$

where

$$u_k^{\text{ff}} = [u_k^{\text{ff}}(0), u_k^{\text{ff}}(1), \ldots, u_k^{\text{ff}}(N-1)]^\top, \tag{2.10}$$

$$y_k = [y_k(1), y_k(2), \ldots, y_k(N)]^\top. \tag{2.11}$$

The matrix G_d and the exogenous signal d repeated at each trial are respectively denoted as

$$G_d = \begin{bmatrix} g_1 & 0 & 0 & \cdots & 0 \\ g_2 & g_1 & 0 & \cdots & 0 \\ g_3 & g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_N & g_{N-1} & g_{N-2} & \cdots & g_1 \end{bmatrix}, \tag{2.12}$$

$$d = [d(1), d(2), \ldots, d(N)]^\top, \tag{2.13}$$

the lifted desired output vector y_d is

$$y_d = \left[y_d(1), y_d(2), \ldots, y_d(N)\right]^\top. \tag{2.14}$$

According to the output form of system (2.9), it can be seen that the performance analysis of the feedback-based ILC system can be transformed into the traditional ILC design objective for the study.

2.3. Problem statement

The achievable performance and flexibility of traditional ILC methods in varying trajectory tasks are limited due to the repetitive requirements of ILC on the system. For this reason, this paper designs a new feedback-based ILC with a 2-D parallel framework. By integrating the feedforward feedback controller into a new feedback controller with learning-based parameters, the experience of repeatedly performing a known task is transferred to a new task to achieve both high tracking performance and flexibility for unknown varying trajectories.

The main design problem of the proposed control optimization scheme is as follows.

Definition 2.1. *The main problem of the proposed control scheme is to design a feedback-based ILC input update law*

$$u_{k+1}^{\text{ff}} = \mathcal{F}(u_k^{\text{ff}}, u_{k-1}^{\text{ff}}, \ldots, e_k, e_{k-1}, \ldots,), \tag{2.15}$$

which is iteratively updated to make the corrected tracking error converges. The LS fitting method is then used to obtain the parameters of the learning-based feedback controller, i.e., $\hat{\theta} = \arg\min_\theta \{\mathcal{V}(\theta)\}$*, which transfers the learning experience from the known task to a brand new task, thus enabling the learning-based feedback controller to respond quickly to varying tasks and achieve the convergence of the system tracking, i.e.,*

$$\lim_{t\to\infty} e(t) \leqslant \epsilon_e, \tag{2.16}$$

where ϵ_e *is a small constant that serves as an upper bound for the tracking error.*

3. Design of feedback-based ILC

In this section, the NOILC algorithm is first introduced to design a suitable ILC updating law to ensure a good tracking performance of the system. Then, the LS fitting algorithm is introduced to obtain a learning-based PID controller.

3.1. ILC algorithm design

In this section, the NOILC algorithm [9] is introduced to optimize the multi-objective performance criteria for each trial, thereby obtaining the optimal input and error information for the feedback-based ILC system. The performance criterion is defined as

$$J_{k+1} \triangleq \|e_{k+1}\|_Q^2 + \|\Delta u_{k+1}^{\text{ff}}\|_R^2 + \|u_{k+1}^{\text{ff}}\|_S^2, \tag{3.1}$$

where

$$e_{k+1} = y_d - G_d u_{k+1}^{\text{ff}} - d, \tag{3.2}$$

$$\Delta u_{k+1}^{\text{ff}} = u_{k+1}^{\text{ff}} - u_k^{\text{ff}}. \tag{3.3}$$

The performance criterion consists of three components: systematic tracking error per trial, input variation between two neighboring trials, and control effort. The role of the Q matrix is weighing error. The purpose of minimizing tracking errors is to efficiently track the desired trajectory, which is the basic goal of ILC. The matrix S is used to weight the feedforward control inputs u_{k+1}^{ff}. By increasing S, it is possible to avoid excessively large control inputs and thus achieve a more stable control effect. The matrix R limits the magnitude of input variations between trials, smoothing

the control action, and thus avoiding large error spikes. The weight matrices Q, R, and S of these three components indicate their priority in the optimization process. By adjusting the values of Q, R, and S, the optimal balance of the system between error, control input variation, and smoothness can be found.

The optimal control inputs can be obtained by minimizing the performance criteria as follows:

$$u_{k+1}^{\mathrm{ff}} = \arg\min_{u_{k+1}^{\mathrm{ff}}} \{J_{k+1}\}. \tag{3.4}$$

Theorem 3.1 (ILC update law)**.** *According to the solution derived from the necessary condition of optimality (3.4), the ILC update rate of the system is*

$$u_{k+1}^{\mathrm{ff}} = L_u u_k^{\mathrm{ff}} + L_e e_k, \tag{3.5}$$

where the gain matrices L_u and L_e are defined as

$$\begin{aligned} L_u &= (G_d^\top Q G_d + R + S)^{-1}(G_d^\top Q G_d + R), \\ L_e &= (G_d^\top Q G_d + R + S)^{-1} G_d^\top Q. \end{aligned} \tag{3.6}$$

Proof. Substituting (3.2) and (3.3) into (3.1) according to the induced norm of the matrix defined in the Hilbert space yields

$$\begin{aligned} J_{k+1} = {} & (y_d - G_d u_{k+1}^{\mathrm{ff}} - d)^\top Q (y_d - G_d u_{k+1}^{\mathrm{ff}} - d) \\ & + (u_{k+1}^{\mathrm{ff}} - u_k^{\mathrm{ff}})^\top R (u_{k+1}^{\mathrm{ff}} - u_k^{\mathrm{ff}}) \\ & + (u_{k+1}^{\mathrm{ff}})^\top S u_{k+1}^{\mathrm{ff}}. \end{aligned} \tag{3.7}$$

Next, find the partial derivative of the performance criterion with respect to u_{k+1}^{ff}. Let $\partial J_{k+1}/\partial u_{k+1}^{\mathrm{ff}} = \mathbf{0}$, which yields

$$R(u_{k+1}^{\mathrm{ff}} - u_k^{\mathrm{ff}}) - G_d^\top Q \left(e_k - G_d(u_{k+1}^{\mathrm{ff}} - u_k^{\mathrm{ff}})\right) + S u_{k+1}^{\mathrm{ff}} = \mathbf{0}. \tag{3.8}$$

Merging the same parts gives

$$(G_d^\top Q G_d + R + S) u_{k+1}^{\mathrm{ff}} - (G_d^\top Q G_d + R) u_k^{\mathrm{ff}} - G_d^\top Q e_k = \mathbf{0}. \tag{3.9}$$

Given that the matrices Q, R, and S are positive definite, it can be demonstrated that $(G_d^\top Q G_d + R + S)$ is invertible. This leads to deriving the ILC update law (3.5), which completes the proof. □

The control objective of this algorithm is to ensure that the system tracking error converges to an acceptable tolerance within a finite number of trials, thereby obtaining the system's optimal control input vector U_κ and error vector E_κ. Specifically, $U_\kappa = [u_\kappa(1), u_\kappa(2), \cdots, u_\kappa(N)]^\top \in \mathbb{R}^{N\times 1}$ and $E_\kappa = [e_\kappa(1), e_\kappa(2), \cdots, e_\kappa(N)]^\top \in \mathbb{R}^{N\times 1}$ represent the control input vector and tracking error vector for the κth trial of the system, where N is the trial length and κ is the maximum number of iterations.

3.2. Least squares error regression

To integrate the feedback plus feedforward controller into a new feedback controller and to enable the transfer of the known task learning experience, this section uses an LS fitting approach to compute the learning-based parameters of an equivalent PID controller based on the optimal input and error sequences of the feedback-based ILC system. Many articles related to system identification or adaptive control discuss the LS identification method, e.g., [21, 27]. Based on the basic idea of linear regression, this section will briefly discuss the method in the PID case.

The PID controller is essentially linear, mainly because the relationship between its control output and the error is linear. Its linear parameterization yields

$$C(\theta, z) = \Psi\theta = \sum_{i=1}^{3} \psi_i \theta_i, \tag{3.10}$$

where Ψ is the basis function vector containing the three discrete-time transfer functions, i.e., $\Psi = [\psi_1(z), \psi_2(z), \psi_3(z)] = [1, \frac{T_s z}{z-1}, \frac{z-1}{T_s z}]$, and the parameter vector $\theta = [\theta_1, \theta_2, \theta_3]^\top = [K_p, K_i, K_d]^\top$. Taking the inverse z-transform each term of (3.10) separately, the output sequence of the discrete PID controller can be expressed as

$$u_{\mathrm{fb}}(j) = \varphi^\top(j)\theta, \tag{3.11}$$

where the information vector $\varphi^\top(j)$ is defined as follows:

$$\varphi^\top(j) = [e(j), T_s \sum_{\tau=0}^{j} e(\tau), (e(j) - e(j-1))/T_s], \; j \in [1, N]. \tag{3.12}$$

According to (3.11) the system feedback output can be written in the following matrix form:

$$U_{\mathrm{fb}} = \phi\theta, \tag{3.13}$$

where

$$U_{\text{fb}} = [u_{\text{fb}}(1), u_{\text{fb}}(2), \cdots, u_{\text{fb}}(N)]^\top, \tag{3.14}$$

$$\phi = \begin{bmatrix} e(1) & T_s \sum_{\tau=0}^{1} e(\tau) & (e(1) - e(0))/T_s \\ \vdots & \vdots & \vdots \\ e(N) & T_s \sum_{\tau=0}^{N} e(\tau) & (e(N) - e(N-1))/T_s \end{bmatrix}. \tag{3.15}$$

The core principle of the LS method is to obtain optimal estimates of the model parameters by minimizing the sum of squares of the residuals between the model output and the actual observed data. In this section, the LS method is employed to minimize the sum of squares of errors between the optimal control input U_κ of the system and the output U_{fb} of the PID controller. This process facilitates the transformation of feedforward and feedback controllers into a unified feedback controller. Based on the observed system data, where $U_\kappa = [u_\kappa(1), u_\kappa(2), \cdots, u_\kappa(N)]^\top$ and $E_\kappa = [e_\kappa(1), e_\kappa(2), \cdots, e_\kappa(N)]^\top$, the parameters of the integrated feedback controller can be determined by minimizing the following quadratic criterion function:

$$\mathcal{V}(\theta) = \sum_{j=1}^{N} v^2(j) = \sum_{j=1}^{N} [u_\kappa(j) - \varphi^\top(j)\theta]^2, \tag{3.16}$$

where $u_\kappa(j)$ is the total input to the feedback-based ILC system, and $\varphi^\top(j)\theta$ is its estimation.

Theorem 3.2. *If $\phi^\top\phi$ is full rank, the optimal $\hat{\theta}$ can be directly calculated by the one-time completion algorithm of LS, that is,*

$$\hat{\theta} = (\phi^\top\phi)^{-1}\phi^\top U_\kappa \tag{3.17}$$

and the error v is the smallest at this time.

Proof. $\mathcal{V}(\theta)$ in (3.16) can be reformulated as

$$\mathcal{V}(\theta) = V^\top V = (U_\kappa - \phi\theta)^\top (U_\kappa - \phi\theta), \tag{3.18}$$

where $\theta = \hat{\theta}$, and $\min \mathcal{V}(\theta) = \mathcal{V}(\hat{\theta})$. Let the partial derivative of $\mathcal{V}(\theta)$ with respect to θ be zero, which yields

$$\left.\frac{\partial \mathcal{V}(\theta)}{\partial \theta}\right|_{\theta=\hat{\theta}} = \left. -2\phi^\top (U_\kappa - \phi\theta)\right|_{\theta=\hat{\theta}} = \mathbf{0}. \tag{3.19}$$

Then,

$$(\phi^\top\phi)\hat{\theta} = \phi^\top U_\kappa. \tag{3.20}$$

Finally, since $\phi^\top\phi$ is full rank, take the inverse of $\phi^\top\phi$ to obtain (3.17), which completes the proof. □

Remark 3.1. *Feedback controllers with learning-based parameters are obtained by LS fitting. The minimization $\mathcal{V}(\theta)$ keeps the fitting error as small as possible, allowing the integrated feedback control system to acquire the approximate input-output characteristics of the original feedforward-feedback combined system for the transfer of the known task learning experience.*

3.3. Learning-based feedback design

Based on the optimal input sequence and the error sequence acquired by the system during the repetitive execution of a certain task, the learning parameter $\hat{\theta}$ of the brand new feedback controller after the integration of the feedforward and feedback controllers is obtained using the LS fitting method. According to the parameter $\hat{\theta}$, a new feedback controller $C(\hat{\theta}, z)$ can be obtained that is as close as possible to the input-output characteristics of the feedback-based ILC system. Then, the output of the learning-based feedback control system can be described as

$$y = \frac{H(z)C(\hat{\theta}, z)}{1 + H(z)C(\hat{\theta}, z)} y_d. \tag{3.21}$$

In addition, the learning-based feedback controller possesses the historical learning experience of ILC for known tasks, and its application to new tasks can realistically provide good tracking performance and robustness.

The specific algorithms are described as shown in Algorithms 1 and 2, which represent the feedback-based ILC design and the learning-based feedback controller parameter calculation, respectively. For Algorithm 1, to analyze the system in the lifting framework, system (2.6) can be converted to (2.9) by solving the impulse response of the corresponding transfer function to obtain the Toeplitz matrix. In Algorithm 2, the feedforward-plus-feedback controller is integrated into a new feedback controller based on the two sequences obtained by Algorithm 1, whose learning-based parameters can be computed to be obtained using the LS fitting method. This approach enhances the flexibility and applicability of ILC in dynamic and uncertain environments by migrating the control experience to a completely new task.

Algorithm 1: ILC algorithm for the feedback-based ILC system.

Input: Reference trajectory y_d; weighting matrices Q, R and S; sampling time N; total number of iterations κ.

Output: Control input $u_k(t)$ and system tracking error $e_k(t)$.

1: **Initialization:** The appropriate initial PID parameters K_p, K_i, and K_d; the initial ILC input u_0^{ff}.
2: Solve for the Toplitz lower triangular matrices G_d and G_r in system (2.9), and then calculate y_0 by substituting u_0^{ff}.
3: Calculate e_0 and then obtain u_1^{ff} according to the designed ILC update law (3.5).
4: **for** k=1,2,$\cdots$, κ **do**
5: Calculate the system output y_k and tracking error e_k according to (2.9).
6: Update the next trial input u_{k+1}^{ff} using ILC input u_k^{ff} and error e_k.
7: **end for**
8: Calculate $u_\kappa^{\text{fb}}(t)$ using the error sequence $e_\kappa(t)$ of the last trial. According to Eq (2.3), combining u_κ^{ff} and $u_\kappa^{\text{fb}}(t)$ yields $u_\kappa(t)$.

Return: $u_\kappa(t)$, $e_\kappa(t)$.

Algorithm 2: Learning-based parameter updating.

Input: Optimal input sequence $u_\kappa(t)$ and error sequence $e_\kappa(t)$.

Output: The learning-based parameters k_p, k_i, and k_d.

1: Give the initial error tracking error $e_\kappa(0)$.
2: **for** j=1,2,$\cdots$, N **do**
3: Calculate the information vector $\varphi^\top(j)$ for each moment based on $e_\kappa(j)$ according to Eq (3.12).
4: **end for**
5: According to Eq (3.17), the learning-based PID controller parameters $\hat{\theta} = [k_p, k_i, k_d]$ are calculated.

4. Convergence and performance analysis

This section first briefly analyzes the stability of the closed-loop feedback system to ensure the effectiveness of the feedback-based ILC design. Then, the convergence and robustness of the proposed algorithm are analyzed.

Prior to the performance analysis, an assumption is given for the subsequent derivations.

Assumption 4.1. *Within the specified finite time interval $t \in [0, N]$, there exists a unique bounded control input $u_d(t)$ such that the output of the feedback-based ILC system accurately tracks the desired trajectory, i.e.,*

$$y_d(t) = G_d u_d(t) + d. \tag{4.1}$$

4.1. Closed-loop feedback system stability

In the control scheme proposed in this paper, selecting an appropriate feedback controller to ensure the stability of the closed-loop system is a prerequisite for designing the ILC optimization algorithm. A stable feedback control system ensures that the system output remains within a bounded range and guarantees that the system stays bounded during the iterative process after incorporating ILC. If the system is unstable, directly introducing ILC may cause the error to diverge during iterations, making it difficult to achieve effective control and learning. In Figure 1, the closed-loop transfer function of the system without considering the feedforward inputs and external disturbances can be expressed as follows:

$$\Phi(z) = \frac{H(z)C(\theta, z)}{1 + H(z)C(\theta, z)}. \tag{4.2}$$

According to (4.2), the characteristic equation of the closed-loop system is given by

$$1 + H(z)C(\theta, z) = 0. \tag{4.3}$$

Whether the system is stable or not depends on the location of the characteristic root, substituting the discrete PID controller expression (2.2) into (4.3) can be obtained as

$$1 + \left(K_p + K_i \frac{T_s z}{z-1} + K_d \frac{z-1}{T_s z}\right) H(z) = 0. \tag{4.4}$$

For system stability, it is necessary to ensure that all poles (i.e., solutions of z) lie within the unit circle, i.e., it satisfies

$$|z_i| < 1, i = 1, 2, \cdots. \tag{4.5}$$

Remark 4.1. *Stability analysis of closed-loop systems can be performed by a variety of methods. For LTI systems, pole analysis, root locus, Nyquist criterion, and Bode plot are commonly used methods.*

4.2. Convergence properties of the proposed algorithm

To facilitate the design of the feedback-based ILC algorithm, this paper adopts the output form of the system as shown in (2.9), which contains the output signals of the feedback control in the course of each trial, so the convergence of the system can be directly analyzed for u_k^{ff}.

Theorem 4.1 (Convergence property)**.** *If the initial states of each trial of the linear discrete system are strictly the same, apply the ILC update law (3.5) to system (2.9). Then, if the condition*

$$\beta = \|L_u - L_e G_d\| \leqslant \xi < 1 \tag{4.6}$$

holds, the tracking error paradigm can converge boundedly, i.e.,

$$\lim_{k\to\infty} \|e_{k+1}\| \leqslant \frac{q_u l}{1-\xi}, \tag{4.7}$$

where $q_u = q u_d$ is a positive constant, and $l = \|G_d\|$. Moreover, the quantity q is defined as a positive scalar as follows:

$$\|I - L_u\| \leqslant q < \left\|G_d^{\top}\right\| \|Q\| \|G_d\| + \|R\| + 1. \tag{4.8}$$

Proof. First, according to Assumption 4.1, the input error δu_k^{ff} and output error δy_k are defined as

$$\begin{cases} \delta u_k^{\mathrm{ff}}(t) = u_d(t) - u_k^{\mathrm{ff}}(t), \\ \delta y_k(t) = y_d(t) - y_k(t), \end{cases} \tag{4.9}$$

where u_d is the desired input to Algorithm 1 rather than the total input to the feedback-based ILC system.

According to (2.9), the tracking error for the kth trial is as follows:

$$\begin{aligned} e_k &= y_d - y_k \\ &= G_d u_d + d - G_d u_k^{\mathrm{ff}} - d \\ &= G_d(u_d - u_k^{\mathrm{ff}}). \end{aligned} \tag{4.10}$$

Combining (4.9) and (4.10) yields

$$e_k = G_d \delta u_k^{\mathrm{ff}}. \tag{4.11}$$

According to (4.9) and (3.5), the input error of the $(k+1)$th batch system is defined as

$$\begin{aligned} \delta u_{k+1}^{\mathrm{ff}} &= u_d - u_{k+1}^{\mathrm{ff}} \\ &= u_d - L_u u_k^{\mathrm{ff}} - L_e e_k \\ &= u_d - L_u(u_d - \delta u_k^{\mathrm{ff}}) - L_e G_d \delta u_k^{\mathrm{ff}} \\ &= (L_u - L_e G_d)\delta u_k^{\mathrm{ff}} + (I - L_u) u_d. \end{aligned} \tag{4.12}$$

This shows the error propagates through a learning gain term $(L_u - L_e G_d)$ and a bias term $(I - L_u)u_d$. If $\|L_u - L_e G_d\| < 1$, the first term decays geometrically. Taking the norm on both sides of (4.12) yields

$$\|\delta u_{k+1}^{\mathrm{ff}}\| \leqslant \|L_u - L_e G_d\| \, \|\delta u_k^{\mathrm{ff}}\| + \|I - L_u\| \|u_d\|. \tag{4.13}$$

Then, prove that $\|I - L_u\|$ is bounded above. Using the compatibility property and the triangle inequality of the norm, an inequality can be established as

$$\|I - L_u\| \leqslant \|L_u\| + 1. \tag{4.14}$$

According to (3.6), extracting the common factor R gives

$$\|L_u\| = \left\| \left(R^{-1}\left(G_d^{\top} Q G_d + S\right) + I\right)^{-1} \left(R^{-1} G_d^{\top} Q G_d + I\right) \right\|. \tag{4.15}$$

Given that the matrices R^{-1}, G_d, Q, R, and S are positive definite matrices, it is obvious that $\left\| \left(R^{-1}\left(G_d^{\top} Q G_d + S\right) + I\right)^{-1} \right\| < 1$. Thus, it is possible to obtain

$$\|L_u\| < \left\| R^{-1} G_d^{\top} Q G_d + I \right\| < \left\| R^{-1} \right\| \left\| G_d^{\top} \right\| \|Q\| \|G_d\| + 1. \tag{4.16}$$

Substitute (4.16) into (4.14) to obtain

$$\|I - L_u\| < \left(\left\| R^{-1} \right\| \left\| G_d^{\top} \right\| \|Q\| \|G_d\| + 1 \right) + 1. \tag{4.17}$$

Subsequently, define a positive scalar q as

$$\|I - L_u\| \leqslant q < \left(\left\| R^{-1} \right\| \left\| G_d^{\top} \right\| \|Q\| \|G_d\| + 1 \right) + 1. \tag{4.18}$$

On the basis of (4.18), equation (4.13) can be represented as

$$\left\| \delta u_{k+1}^{\mathrm{ff}} \right\| \leqslant \|L_u - L_e G_d\| \left\| \delta u_k^{\mathrm{ff}} \right\| + q \|u_d\|. \tag{4.19}$$

Let $q_u = q\|u_d\|$, according to the inequality relation, after k trials, there exists

$$\|\delta u_{k+1}^{\mathrm{ff}}\| \leqslant \|L_u - L_e G_d\|^k \, \|\delta u_0^{\mathrm{ff}}\| + \frac{1 - \|L_u - L_e G_d\|^k}{1 - \|L_u - L_e G_d\|} q_u. \tag{4.20}$$

If condition (4.6) holds, when the number of experiments k tends to infinity, it follows that

$$\lim_{k\to\infty} \|L_u - L_e G_d\|^k = 0. \tag{4.21}$$

Then, the input error can be expressed as

$$\lim_{k\to\infty} \delta u_{k+1}^{\mathrm{ff}} \leqslant \frac{q_u}{1-\xi}. \tag{4.22}$$

Combining (4.11) and (4.22) gives

$$\lim_{k\to\infty} e_{k+1} = \lim_{k\to\infty} G_d \delta u_{k+1}^{\mathrm{ff}} \leqslant \frac{q_u \|G_d\|}{1-\xi}. \tag{4.23}$$

Finally, let $l = \|G_d\|$ to obtain

$$\lim_{k\to\infty} e_{k+1} \leqslant \frac{q_u l}{1-\xi}, \tag{4.24}$$

which is the convergence of the tracking error paradigm to a bounded value, and the proof is complete. □

Remark 4.2. *In particular, when S is a zero matrix and $L_u = I$, the system tracking error can achieve perfect tracking, i.e.,* $\lim_{k\to\infty} \|e_{k+1}\| = 0$.

Based on the above analysis, the feedback-based ILC system can achieve excellent tracking performance. By using the system's optimal control input and error signals, the LS method is applied to minimize the error between the optimal input and the PID model output, thereby integrating the feedforward and feedback controllers into a feedback controller with learning-based parameter $\hat{\theta} = [k_p, k_i, k_d]$. With a sufficiently small fitting error, a learning-based feedback control system can transfer the learning experience from repeated performance of a task to a new task, and achieve efficient tracking of the changing trajectory task without re-learning, i.e., $\lim_{t\to\infty} e(t) \leqslant \epsilon_e$. Substituting the parameter $\hat{\theta}$ into (4.4) to solve for the closed-loop poles, the system convergence can be further analyzed based on the distribution of the system pole positions.

4.3. Robustness analysis

In practical applications, the dynamics of the controlled system may deviate from its nominal model due to modeling inaccuracies or parameter variations. To this end, the robustness of the proposed algorithm under model uncertainty is investigated. Introducing a bounded modeling error factor Δ_G, the system model G_Λ with multiplicative uncertainty is defined as

$$G_\Lambda = G_d(I + \Delta_G), \tag{4.25}$$

where $\Delta_G = W\Lambda$, W is the weighting matrix, and Λ is the uncertainty factor satisfying $\|\Lambda\| \leq 1$. According to Theorem 3.1, a sufficient condition for robust convergence of the uncertain system (4.25) can be derived similarly as follows:

$$\|L_u - L_e G_\Lambda\| < 1. \tag{4.26}$$

Using the expression for the nominal operator in (3.6) yields

$$L_u - L_e G_d = (G_d^\top Q G_d + R + S)^{-1} R, \tag{4.27}$$

which suggests that the weight matrix R significantly affects the convergence behavior.

Theorem 4.2. *When the weighting matrix $R = rI$ and satisfies $r \geq 0$, the sufficient condition for robust convergence of the uncertain controlled system under the action of the optimal control law (3.5) is*

$$\left\| \left(G_d^\top Q G_d + S\right)^{-1} G_d^\top Q G_d W \right\| < 1. \tag{4.28}$$

Proof. Combining (3.6) and (4.25) yields

$$\begin{aligned} \|L_u - L_e G_\Lambda\| &= \left\| \left(G_d^\top Q G_d + R + S\right)^{-1} (R - G_d^\top Q G_d \Delta_G) \right\| \\ &\leq \left\| \left(G_d^\top Q G_d + R + S\right)^{-1} G_d^\top Q G_d W \right\| \|\Lambda\| \\ &\quad + \left\| \left(G_d^\top Q G_d + R + S\right)^{-1} R \right\|, \end{aligned} \tag{4.29}$$

when $R = \mathbf{0}$, the second term vanishes, and condition (4.28) directly ensures the robust convergence condition (4.26). Since the matrix $G_d^\top Q G_d + S$ is positive-definite symmetric, a singular value decomposition of it yields

$$G_d^\top Q G_d + S = X \Sigma X^\top, \tag{4.30}$$

where X is a unitary matrix and Σ is a full rank diagonal matrix with diagonal elements σ_i, denoted by $\Sigma = \mathrm{diag}\{\sigma_i\}$. To simplify the calculation, let

$$Z \triangleq \left(X \Sigma X^\top\right)^{-1} G_d^\top Q G_d W. \tag{4.31}$$

Suppose there exists a constant α, $0 < \alpha < 1$, such that $\|Z\| = \alpha < 1$ and $\|Z\Lambda\| \leq \alpha$. Then, a further derivation based

on (4.29)–(4.31) yields

$$\begin{aligned}
&\max_{\Delta}\left\|\left(G_d^{\top}QG_d+R+S\right)^{-1}\left(R-G_d^{\top}QG_d\Delta_G\right)\right\| \\
&=\max_{\Delta}\left\|\left(X\Sigma X^{\top}+rI\right)^{-1}\left(rI+X\Sigma X^{\top}Z(-\Lambda)\right)\right\| \\
&\leq\left\|\left(X\Sigma X^{\top}+rI\right)^{-1}\left(rI+\alpha X\Sigma X^{\top}\right)\right\| \\
&=\left\|X(\mathbf{\Sigma}+rI)^{-1}\left(rI+\alpha\mathbf{\Sigma}\right)X^{\top}\right\| \\
&=\left\|(\mathbf{\Sigma}+rI)^{-1}(rI+\alpha\mathbf{\Sigma})\right\| \\
&=\max_{i}\frac{\alpha\sigma_i+r}{\sigma_i+r}<1,\forall r\in\mathbb{N}.
\end{aligned} \tag{4.32}$$

From this, it can be introduced that when R=rI, the robust convergence condition of the uncertainty system (4.25) is $\left\|\left(G_d^{\top}QG_d+S\right)^{-1}G_d^{\top}QG_dW\right\| < 1$, and the proof is completed. □

Remark 4.3. *The robustness condition in (4.26) reflects the system's tolerance to modeling errors. In practice, the uncertainty factor* Λ *may increase due to abrupt load changes, unmodeled dynamics, or sensor noise. If the weighting matrix R is too small (i.e., control effort is weakly penalized), the term* $\|L_u - L_eG_\Lambda\|$ *may exceed* 1*, leading to divergence of the learning process. Therefore, increasing R or selecting an appropriate weighting matrix W can improve robustness by effectively suppressing the impact of model uncertainty.*

5. Simulation verification

To illustrate the applicability and effectiveness of the proposed method, a digital model of a DC motor with constant armature current is used [28]. An example is a position servo control system. The rotation angle of the DC motor is controlled by an electric field winding, whereby the motor rotation angle is controlled by adjusting the voltage to output the desired speed. Simulation results validate the effectiveness of the proposed algorithm and comparing it with other algorithms also proves the superior performance of the algorithm.

5.1. Simulation specification

A DC motor system with a constant armature current is illustrated in Figure 3. The state variables are defined as $x(t) = \left[i_f(t) \quad \omega(t) \quad \vartheta(t)\right]^{\top}$, where $\omega(t)$ is angular velocity and $\vartheta(t)$ is the angle of the DC motor. Moreover, V_f is the driving voltage that directly controls the motor position. The input variable of the system is defined as $u(t) = V_f(t)$, while the output variable is expressed as $y(t) = \vartheta(t)$. The DC motor system's state space model matrix can be obtained as follows:

$$A=\begin{bmatrix}-\frac{R_f}{L_f} & 0 & 0\\ \frac{k_m}{J} & -\frac{f}{J} & 0\\ 0 & 1 & 0\end{bmatrix}, B=\begin{bmatrix}\frac{1}{L_f}\\ 0\\ 0\end{bmatrix}, C=\begin{bmatrix}0 & 0 & 1\end{bmatrix}, \tag{5.1}$$

where the specific parameter values are shown in Table 1.

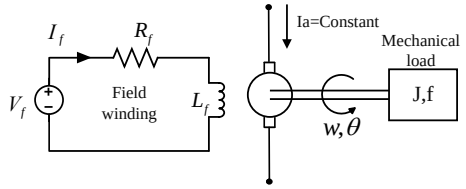


Figure 3. DC motor with constant armature current.

Table 1. A detailed delineation of the system parameters.

variable	definition	value
R_f	field winding resistance	15 Ω
L_f	inductance	1 H
k_m	motor torque ratio	0.5 Nm/A
J	mechanical load inertia momentum	2 Nm/A
f	friction ratio	2 Nm/A

The model of the DC motor position control is a linear system with single input and single output. To ensure that the system output adheres to a specified desired trajectory, it is necessary to discretize the motor model using a zero-order keeper with a sampling time of $T_s = 0.01\ s$. The discrete-time state-space form of the system is obtained as

$$\begin{cases}x_k(t+1)=A_dx_k(t)+B_du_k(t),\\ y_k(t)=C_dx_k(t),\end{cases} \tag{5.2}$$

where the system state matrix, the input-output matrix is

$$A_d = \begin{bmatrix} 0.8187 & 0 & 0 \\ 0.4526 & 0.9975 & 0 \\ 0.0023 & 0.0100 & 1 \end{bmatrix}, \quad B_d = \begin{bmatrix} 0 \\ 0.0197 \\ 0.0211 \end{bmatrix}, C_d = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}. \tag{5.3}$$

The expected output trajectory is defined as a time-varying parabolic function: $y_d(t) = 1.25 \times t(T - t)$, where T is the trial duration here set to 10 s. Applying the proposed algorithm to a DC motor position control system can achieve good tracking performance, where the feedback controller adjusts the motor voltage to control the motor position based on the real-time error, while the ILC continuously learns from its history during repetitive operations to reduce the error. Specific performance is analyzed below.

5.2. *Performance of proposed algorithm*

Simulation experiments are conducted by applying the proposed algorithm to the DC motor position control system. The tracking performance of the system is evaluated using the root mean square error (RMSE), which is defined as $\text{RMSE}(k) = \sqrt{\sum_{t=0}^{N} |e_k(t)|^2/(N+1)}$. First, to ensure that the closed-loop feedback system is stable, the initial PID controller parameters are selected as $K_p = 0.2, K_i = 0.1, and\ K_d = 0.05$ using the traditional Ziegler–Nichols (ZN) regulation method. The proposed Algorithm 1 is then applied to the known task of DC motor position system (5.2) for a total of 20 trials with weight matrices of $Q = I$, $R = 0.1I$, and $S = 0.01I$. The simulation results are shown in Figures 4 and 5.

Figure 4 shows the desired trajectory of the system, the initial feedback control output trajectory, and the output trajectory of the system after incorporating ILC, respectively. Figure 5 shows the RMSE curve of the feedback-based ILC system over successive trials. As can be seen from the figures, the proposed method exhibits robust tracking performance. With only a small number of iterations, the system output is able to track the desired trajectory almost accurately, with a stable convergence of the RMSE to a constant value of about 0.012. After the feedback-based ILC system converges, the feedback plus feedforward controller is integrated into a new feedback controller. The parameters of this controller are obtained using Algorithm 2 and applied to the DC motor system to track different task trajectories. The simulation results are shown in Figure 6.

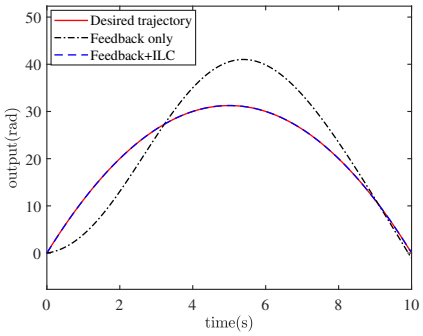


Figure 4. System desired trajectory and output trajectory before and after adding ILC.

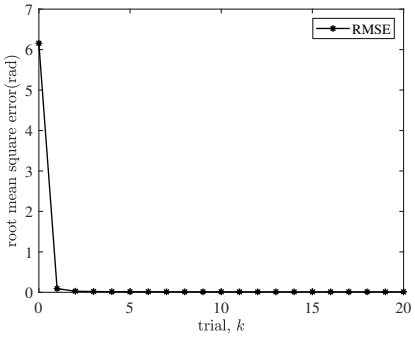


Figure 5. Convergence performance of the RMSE of the proposed method.

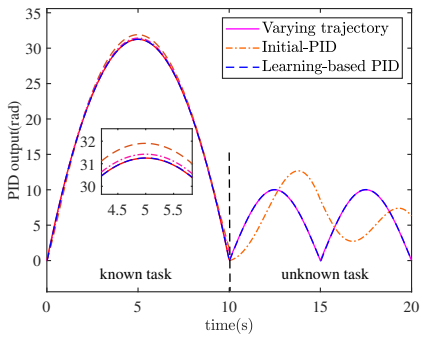


Figure 6. Learning-based PID control system output trajectory.

In the figure, the red solid lines represent the desired trajectories for different tasks, with the unknown task trajectory defined as $y_{r_1}(t) = |10\sin(0.2\pi t)|$. The blue dashed line represents the system output under the learning-based PID controller. As illustrated in Figure 6, the system not only achieves good tracking of the known task trajectory after a finite number of learning iterations, but also can effectively track the varying trajectory without re-learning, showing excellent generalization ability and control performance.

To further assess the robustness of the proposed control method, two types of disturbances are considered: random noise and real-time perturbations. First, random noise with a standard deviation of 0.2 is added to the output signals of both the feedback-based ILC system and the learning-based feedback control system. The simulation results are presented in Figure 7. As observed, the presence of stochastic noise slightly affects the tracking performance. However, with the progression of iterations, the system gradually adapts to the non-repetitive disturbances. The final output trajectory is able to follow the desired trajectory. The learning-based PID feedback system also suppresses random disturbances well, thanks to the ILC's guided learning and the PID controller's robustness. Next, the ability of the proposed method to handle real-time disturbances is evaluated. Specifically, at the 10th trial of the feedback-based ILC system, a unit step disturbance with an amplitude of 1 is introduced at $t = 3$ s. The corresponding simulation results are depicted in Figure 8.

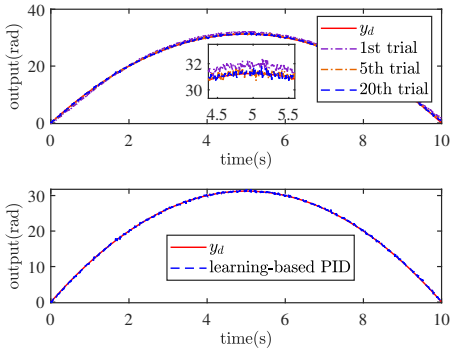


Figure 7. Feedback-based ILC system output and feedback system output under noise interference.

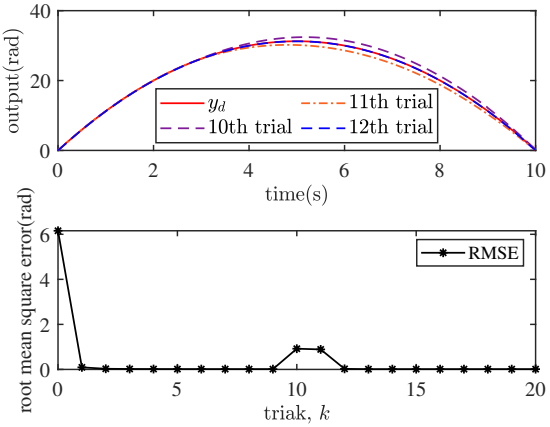


Figure 8. System output trajectories and root mean square error curves of the 10th, 11th, and 12th trials under real-time step disturbance.

Although the real-time perturbation causes the system output to deviate from the desired value, the error is effectively compensated within the current trial under the effect of feedback control, and the system output gradually converges to the expected trajectory as the iteration proceeds. The results show that the proposed method can effectively suppress real-time disturbances.

Finally, the efficacy of the proposed algorithm against model uncertainty is verified. To reflect practical conditions such as load variations and parameter drifts in motor systems, a modeling uncertainty term Δ_G is introduced. The uncertainty factor $\|\Lambda\| = 0.67$ is selected, and the weight matrix $\|W\| = 0.53$ is set so that it satisfies condition (4.26). Then, the uncertain system is simulated using the same control strategy as that of the nominal system, and the results are shown in Figure 9. From the figure, it can be seen that compared with the nominal system, the control performance of the uncertain system is slightly degraded, but it still achieves a small tracking error after a finite number of trials. Moreover, although different weight matrices R affect the convergence speed of the system error, they do not affect the overall convergence of the system to the model uncertainty. These results verify the good adaptability and robustness of the proposed method to model uncertainty.

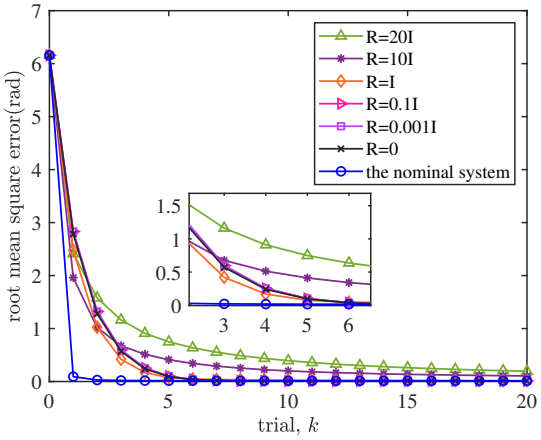


Figure 9. Root mean square error tracking curves of the system at each trial under model uncertainty.

axis. This synergy results in improved convergence speed and robustness to uncertainties.

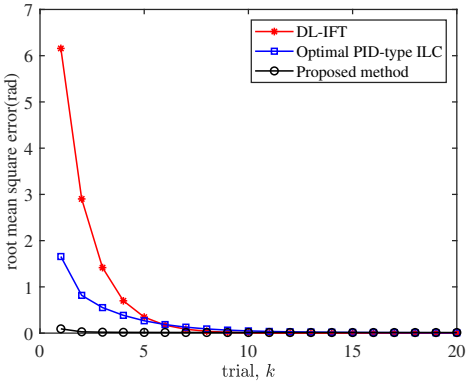


Figure 10. Comparison of the RMSE curves for the three control methods.

5.3. *Compared With conventional counterparts*

To provide further evidence of the effectiveness and benefits of the proposed method, two additional methods are implemented in the DC motor system (5.2) for simulation comparisons. One is the dual-loop iterative feedback tuning (DL-IFT) method proposed in [29], an improved version of the classic IFT algorithm that accelerates convergence by simultaneously updating the inner and outer loop parameters based on measured data. Another approach is the norm optimal PID-type ILC [28], where the PID parameters are continuously updated during the iteration process to find the optimal solution, where $Q = 0.1I$ and $R = I$, the initial PID parameters are the same as the initial values chosen in this paper.

Figure 10 shows the RMSE curves of the tracking errors obtained by these three methods. As seen, all algorithms achieve convergence within a few iterations. However, the proposed method exhibits faster convergence speed compared to both DL-IFT and norm-optimal PID-type ILC. Additionally, it consistently achieves lower tracking error across most of the time steps, indicating superior control accuracy. The enhanced performance of the proposed method can be attributed to its integrated structure, which combines feedback control and iterative learning. The feedback component enables real-time suppression of time-domain disturbances, while the ILC mechanism refines control input iteratively using the error along the iteration

Finally, to further evaluate the robust adaptability of the learning-based PID controller obtained through the proposed framework, it is compared with PID controllers designed using the ZN tuning method and the DL-IFT algorithm. Figure 11 shows the output response curves of the three controllers under varying reference trajectories, where the known task trajectory is y_d and the unknown task trajectory is denoted as

$$y_{r_2}(t) = \begin{cases} 10(t-10), & t \in [10, 13] \\ 30, & t \in [13, 15] \\ 30 + 10(t-15), & t \in [15, 18] \\ 60, & t \in [18, 20] \end{cases}. \tag{5.4}$$

The simulation results indicate that the learning-based PID controller exhibits stronger adaptability to set-point changes and achieves better tracking performance under varying tasks.

In summary, the proposed algorithm not only performs well in terms of control accuracy and convergence speed, but also exhibits excellent robustness and adaptability in the presence of model uncertainties and trajectory variations. The PID controller obtained by fitting the optimal input-output data of the combined feedback and ILC system inherits the dynamic characteristics of the original system and can effectively handle trajectory tracking tasks with unknown changes in the time domain. The effectiveness and

superiority of the algorithm are verified through the above simulations.

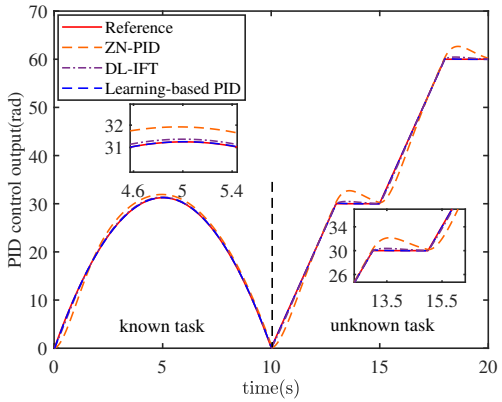


Figure 11. PID output trajectories of three different methods for variable trajectory task.

6. Conclusions and future work

This paper proposes a new ILC scheme for linear systems with repetitive characteristics to track varying reference trajectories. First, based on the known task, a feedback-based ILC algorithm is designed to achieve high tracking performance by combining inter-trial iterative learning and intra-trial real-time feedback to effectively eliminate the error caused by system uncertainty. To apply the learning experience of repeatedly performing a known task to a new task, the optimal parameters of the learning-based feedback controller are estimated by LS fitting based on the optimal output sequence and error sequence of the system. Through this process, the feedback plus feedforward controller is integrated into a new learning-based feedback controller that can achieve effective tracking of changing task trajectories without limiting its duration. Simulation results show that the proposed method has good tracking performance and robustness, and it improves the applicability and flexibility of the ILC control method for varying tasks.

Future work will focus on include experimental validation to further explore the practical performance of the proposed method and investigate its potential application in nonlinear systems.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under Grant 62361136585, partially by the 111 Project under Grant B23008, partially by the National Science Centre in Poland under Grant 2023/48/Q/ST7/00205, and partially by the Serbian Ministry of Science, Technological Development and Innovation under Grant 451-03-137/2025-03/200108.

Conflict of interest

All authors declare that there are no conflicts of interests in this paper.

References

1. S. Arimoto, S. Kawamura, F. Miyazaki, Bettering operation of robots by learning, *J. Robot. Syst.*, **1** (1984), 123–140. https://doi.org/10.1002/rob.4620010203
2. J. Lu, Z. Cao, Q. Hu, Z. Xu, W. Du, F. Gao, Optimal iterative learning control for batch processes in the presence of time-varying dynamics, *IEEE Trans. Syst. Man Cybern.: Syst.*, **52** (2022), 680–692. https://doi.org/10.1109/TSMC.2020.3031669
3. C. E. Boudjedir, M. Bouri, D. Boukhetala, Model-free iterative learning control with nonrepetitive trajectories for second-order mimo nonlinear systems—application to a delta robot, *IEEE Trans. Ind. Electron.*, **68** (2021), 7433–7443. https://doi.org/10.1109/TIE.2020.3007091
4. R. Zhou, C. Hu, Z. Wang, Y. Zhu, M. Tomizuka, Real-time iterative compensation control using plant-injection feedforward architecture with application to ultraprecision wafer stages, *IEEE Trans. Ind. Inf.*, **20** (2024), 11708–11719. https://doi.org/10.1109/TII.2024.3413294

5. K. Qian, Z. Li, S. Chakrabarty, Z. Zhang, S. Q. Xie, Robust iterative learning control for pneumatic muscle with uncertainties and state constraints, *IEEE Trans. Ind. Electron.*, **70** (2023), 1802–1810. https://doi.org/10.1109/TIE.2022.3159970
6. D. A. Bristow, M. Tharayil, A. G. Alleyne, A survey of iterative learning control, *IEEE Control Syst. Mag.*, **26** (2006), 96–114. https://doi.org/10.1109/MCS.2006.1636313
7. H. S. Ahn, Y. Q. Chen, K. L. Moore, Iterative learning control: brief survey and categorization, *IEEE Trans. Syst., Man, Cybern. C*, **37** (2007), 1099–1121. https://doi.org/10.1109/TSMCC.2007.905759
8. D. Shen, Y. Wang, Survey on stochastic iterative learning control, *J. Process Control*, **12** (2014), 64–77. https://doi.org/10.1016/j.jprocont.2014.04.013
9. K. L. Barton, A. G. Alleyne, A norm optimal approach to time-varying ilc with application to a multi-axis robotic testbed, *IEEE Trans. Control Syst. Technol.*, **19** (2011), 166–180. https://doi.org/10.1109/TCST.2010.2040476
10. R. Chi, Z. Hou, S. Jin, A data-driven adaptive ilc for a class of nonlinear discrete time systems with random initial states and iteration-varying target trajectory, *J. Frankl. Inst.*, **352** (2015), 2407–2424. https://doi.org/10.1016/j.jfranklin.2015.03.014
11. Z. Zhuang, H. Tao, Y. Chen, V. Stojanovic, W. Paszke, An optimal iterative learning control approach for linear systems with nonuniform trial lengths under input constraints, *IEEE Trans. Syst. Man, Cybern.: Syst.*, **53** (2023), 3461–3473. https://doi.org/10.1109/TSMC.2022.3225381
12. C. Zhou, L. Jia, J. Li, Y. Chen, Data-driven two-dimensional integrated control for nonlinear batch processes, *J. Process Control*, **135** (2024), 103160. https://doi.org/10.1016/j.jprocont.2023.103160
13. I. Chin, S. J. Qin, K. S. Lee, M. Cho, A two-stage iterative learning control technique combined with real-time feedback for independent disturbance rejection, *Automatic*, **40** (2004), 1913–1922. https://doi.org/10.1016/j.automatica.2004.05.011
14. Y. Wang, H. Zhang, S. Wei, D. Zhou, B. Huang, Control performance assessment for ilc-controlled batch processes in a 2-d system framework, *IEEE Trans. Syst. Man, Cybern.: Syst.*, **48** (2018), 1493–1504. https://doi.org/10.1109/TSMC.2017.2672563
15. C. Chen, Z. Xiong, Y. Zhong, Design and analysis of integrated predictive iterative learning control for batch process based on two-dimensional system theory, *Chin. J. Chem. Eng.*, **22** (2014), 762–768. https://doi.org/10.1016/j.cjche.2014.05.008
16. L. Zhou, L. Jia, Y. L. Wang, D. Peng, W. Tan, An integrated robust iterative learning control strategy for batch processes based on 2d system, *J. Process Control*, **85** (2020), 136–148. https://doi.org/10.1016/j.jprocont.2019.11.011
17. X. Liu, L. Ma, X. Kong, K. Y. Lee, Robust model predictive iterative learning control for iteration-varying-reference batch processes, *IEEE Trans. Syst., Man, Cybern.: Syst.*, **51** (2021), 4238–4250. https://doi.org/10.1109/TSMC.2019.2931314
18. L. Song, J. Shi, Adaptive pi control of ultrasonic motor using iterative learning methods, *ISA Trans.*, **139** (2023), 499–509. https://doi.org/10.1016/j.isatra.2023.03.032
19. R. Chi, H. Li, D. Shen, Z. Hou, B. Huang, Enhanced p-type control: indirect adaptive learning from set-point updates, *IEEE Trans. Autom. Control*, **68** (2023), 1600–1613. https://doi.org/10.1109/TAC.2022.3154347
20. G. Sebastian, Y. Tan, D. Oetomo, Convergence analysis of feedback-based iterative learning control with input saturation, *Automatica*, **101** (2019), 44–52. https://doi.org/10.1016/j.automatica.2018.11.045
21. K. K. Tan, S. Zhao, K. Y. Chua, W. K. Ho, W. W. Tan, Iterative learning approach toward closed-loop automatic tuning of pid controllers, *Ind. Eng. Chem. Res.*, **45** (2006), 4093–4100. https://doi.org/10.1021/ie060093e
22. T. Liu, X. Z. Wang, J. Chen, Robust PID based indirect-type iterative learning control for batch processes with time-varying uncertainties, *J. Process Control*, **24** (2014), 95–106. https://doi.org/10.1016/j.jprocont.2014.07.002

23. S. Hao, T. Liu, E. Rogers, Extended state observer based indirect-type ILC for single-input single-output batch processes with time-and batch-varying uncertainties, *Automatica*, **112** (2020), 108673. https://doi.org/10.1016/j.automatica.2019.108673

24. M. Li, J. Xiong, R. Cheng, Y. Zhu, K. Yang, F. Sun, Rational feedforward tuning using variance-optimal instrumental variables method based on dual-loop iterative learning control, *IEEE Trans. Ind. Inf.*, **19** (2023), 2585–2595. https://doi.org/10.1109/TII.2022.3166590

25. N. Amann, D. H. Owens, E. Rogers, Iterative learning control using optimal feedback and feedforward actions, *Int. J. Control*, **65** (1996), 277–293. https://doi.org/10.1080/00207179608921697

26. J. V. Zundert, J. Bolder, T. Oomen, Optimality and flexibility in iterative learning control for varying tasks, *Automatica*, **67** (2016), 295–302. https://doi.org/10.1016/j.automatica.2016.01.026

27. H. Liu, Y. Li, Y. Zhang, Y. Chen, Z. Song, Z. Wang, et al., Intelligent tuning method of pid parameters based on iterative learning control for atomic force microscopy, *Micron*, **104** (2018), 26–36. https://doi.org/10.1016/j.micron.2017.09.009

28. F. Memon, C. Shao, An optimal a pproach to online tuning method for pid type iterative learning control, *Int. J. Control Autom. Syst.*, **18** (2020), 1926–1935. https://doi.org/10.1007/s12555-018-0840-0

29. M. Li, Y. Zhu, K. Yang, L. Yang, C. Hu, H. Mu, Convergence rate oriented iterative feedback tuning with application to an ultraprecision wafer stage, *IEEE Trans. Ind. Electron.*, **66** (2019), 1993–2003. https://doi.org/10.1109/TIE.2018.2838110

AIMS Press