

# Implementation of iterative learning control in a walking beam motion system with a flexible coupled robotic mechanism

Dragan Kostić<sup>1</sup>, Milan Matijević<sup>2</sup>, Vojislav Filipović<sup>3</sup>  
and Saša Ćuković<sup>4</sup>

## Abstract

This paper focuses on the high-precision engineering application of designing and implementing a norm-optimal iterative learning control (ILC) algorithm for the precise positioning of a robotic gripper tip flexibly coupled to an industrial walking beam system. The ILC algorithm generates a feedforward or exogenous signal to minimize tracking errors, thereby enabling high-precision motion control. However, integrating the ILC signal into an existing feedback control system can introduce control signal discontinuities and fluctuations, potentially impairing performance.

To address these challenges, this paper proposes structural adjustments to the feedback control algorithm and the implementation of the ILC signal. The primary objectives are to prevent controller fragility caused by the integration of the ILC signal, ensure stable and feasible dynamics of the control variable, and maintain the desired system performance. This work demonstrates the novel application of norm-optimal ILC to a walking beam system with a flexibly coupled robotic mechanism. To the best of our knowledge, this specific application has not been explored in the literature. Validation through simulations, conducted using a nonlinear plant model in Simscape/Simulink, provides valuable insights into the dynamic behavior and control performance of the system.

## Keywords

iterative learning control, serial norm-optimal ILC, flexible multibody dynamics, data-driven motion control, controller fragility, control dynamics, ILC implementation

Date received: 13 December 2024; accepted: 28 July 2025

Handling Editor: Divyam Semwal

## Introduction

Iterative Learning Control (ILC) is a high-performance control technique tailored for systems executing repetitive tasks. It enhances tracking accuracy by utilizing error information from previous iterations or trials to adjust future control inputs, much like a basketball player improving their free throw through practice.<sup>1–3</sup> ILC is particularly effective for systems that demand precise tracking of predefined trajectories, as it progressively reduces tracking errors over successive repetitions.<sup>4,5</sup> Moreover, it maintains accurate reference

<sup>1</sup>Enabling Technology Mechatronics, ASMPT Center of Competency, Beuningen, The Netherlands

<sup>2</sup>Centre for Applied Automatic Control, Faculty of Engineering at University of Kragujevac, Serbia

<sup>3</sup>Department for Automatic Control, Robotic and Fluid Technique, Faculty of Mechanical and Civil Engineering, University of Kragujevac, Kraljevo, Serbia

<sup>4</sup>Department for Health Sciences and Technology—D-HEST, Swiss Federal Institute of Technology—ETH Zurich, Switzerland

### Corresponding author:

Saša Ćuković, Department for Health Sciences and Technology—D-HEST, Swiss Federal Institute of Technology—ETH Zurich, Gloriastrasse 37/39, Zurich 8092, Switzerland.  
Email: cukovics@ethz.ch



Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (<https://creativecommons.org/licenses/by/4.0/>) which permits any use, reproduction and distribution of the work

without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (<https://us.sagepub.com/en-us/nam/open-access-at-sage>).

tracking even in the presence of incomplete system models or recurring but unknown disturbances and initial conditions across iterations.<sup>1,5</sup>

Although the concept of Iterative Learning Control (ILC) is relatively recent, its origins date back to the 1970s.<sup>6,7</sup> The first formal and mathematically rigorous study was conducted by Arimoto et al.,<sup>8</sup> which is widely regarded as the foundation of ILC research. The first comprehensive book on ILC, authored by Moore,<sup>9</sup> focused on deterministic systems. Thirty years later, Rogers et al.<sup>10</sup> presented an extensive discussion covering ILC applications across linear, nonlinear, distributed, and stochastic systems. For a thorough overview of the fundamental principles, applications, and recent developments in ILC, the review papers<sup>2-4,11-13</sup> provide excellent resources.

Based on the literature, three main approaches to synthesizing ILC algorithms are outlined by Matijevic and Kostic<sup>5</sup>: (1) heuristic methods, (2) model inversion-based approaches for feedback systems or controlled subsystems, and (3) optimization-based formulations. Among these, optimization-based designs have demonstrated broad applicability across various domains.<sup>14,15</sup> Notable examples include industrial robots, pick-and-place machines, electron microscopes, free-electron lasers, printing systems, wafer stages, healthcare technologies, broiler production systems, quantum control, nuclear fusion, additive manufacturing, and servo systems.

Regardless of the synthesis method, most ILC strategies focus on tracking a reference signal defined over a finite time interval, which is particularly relevant for high-precision engineering tasks. As a result, the spatial tracking problem has also been approached using ILC techniques. However, current implementations of spatial ILC remain highly application-specific, and a general framework applicable across a broader class of systems has yet to be established.<sup>16</sup>

In the work of Saab et al.,<sup>17</sup> ILC was applied to industrial robot manipulators, with emphasis on challenges related to initial state mismatches and model uncertainties. To address this, a specialized homing controller was introduced to automate state initialization at the end of each iteration. Despite this advancement, transient vibrations were observed, indicating the need for further refinement. Similarly, Yan et al.<sup>18</sup> investigated ILC for robotic manipulators operating under complex initial conditions.

The application of ILC to robotic systems with control variable constraints is addressed in several studies. Meng and He<sup>19</sup> investigated the use of ILC for a robotic arm with input constraints, focusing on vibration suppression and accurate trajectory tracking. Input limitations such as actuator saturation and dead zones can introduce undesirable vibrations, compromising both performance and stability. Chen et al.<sup>20</sup>

further explored ILC design under constraints on both control and output variables, demonstrating its effectiveness in maintaining tracking accuracy despite these challenges.

In industrial settings where computational resources are limited, Chi et al.<sup>21</sup> proposed a computationally efficient ILC implementation that avoids matrix inversion. This approach significantly reduces the computational burden, making it practical even for applications with a large number of ILC signal samples.

Meng and He<sup>22</sup> presented an ILC framework aimed at addressing practical implementation challenges in flexible structures, such as vibration suppression, input saturation, dead zones, backlash, external disturbances, and trajectory tracking. The book also introduces simplified partial differential equation models to describe typical problems in flexible structures, which are generally more difficult to stabilize using ILC.

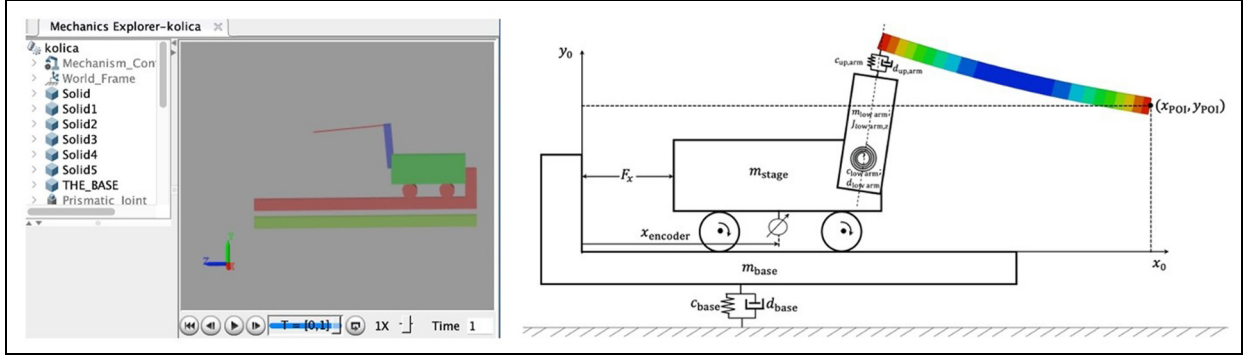
Rogers et al.<sup>10</sup> identified the phenomenon of *long-term performance degradation* in ILC implementations—a situation where the tracking error initially decreases across iterations but begins to grow again after many trials. This is interpreted as an implementation-related issue rather than a limitation of the theoretical framework.

The design of constrained ILC, including approaches for handling input saturation, is also discussed, notably through the use of receding horizon techniques.

Rogers et al.<sup>23</sup> highlight the numerous opportunities for advancing ILC, particularly in the areas of design, application, and implementation. Persistent challenges—such as actuator saturation, backlash, and integrator windup—underscore the need for continued research to enhance its practical applicability.

This paper investigates the application of norm-optimal ILC to a walking beam motion system featuring a flexibly coupled robotic mechanism. It addresses key implementation challenges and proposes modifications to the control algorithm to enable successful deployment. To the best of our knowledge, the use of norm-optimal ILC in walking beam motion systems has not been reported in existing literature. Related aspects of applying norm-optimal ILC in mechatronic systems with flexible couplings are discussed in References 22, 24–26.

Previous research on walking beam systems has primarily focused on classical control architectures, including PLC-based logic,<sup>27</sup> finite-element-based load simulations,<sup>28</sup> nonlinear model predictive control,<sup>29</sup> and multivariable structure design.<sup>30</sup> These approaches require accurate modeling and often lack adaptability to time-varying system dynamics or unmodeled disturbances. The use of iterative learning control (ILC), especially in its norm-optimal form, has not yet been reported for walking beam applications, despite the



**Figure 1.** Walking beam motion system with a flexible coupled robotic mechanism.

method's proven strengths in repetitive, high-precision motion systems.

Walking beam systems are widely used transfer mechanisms in manufacturing environments, particularly in machining-type transfer lines.<sup>31</sup> They have diverse industrial applications, including bonding machines, metal forming lines, packaging processes that require precise placement, and coordination with pick-and-place units or robotic arms. Their motion is synchronized with upstream and downstream equipment—such as conveyor belts, stamping machines, and robotic grippers—to ensure high throughput and coordinated operation. By synchronously lifting, advancing, and positioning parts across workstations, walking beams enhance automation and productivity.

Over time, these systems have evolved from basic transport components into sophisticated tools for precision motion and assembly tasks. However, achieving both high speed and high positioning accuracy remains a challenge, especially in mitigating vibration. Iterative Learning Control (ILC) addresses these challenges by learning from previous executions to compensate for system uncertainties and improve tracking accuracy.<sup>32</sup> It generates feedforward signals or exogenous inputs that reduce tracking errors, even when system models are incomplete.

Despite these benefits, implementing ILC may lead to abrupt changes in control variables, such as discontinuities or fluctuations, that degrade overall system performance. To mitigate these issues, adjustments to both the control algorithm structure and the implementation of the ILC signals are necessary.

This paper proposes modifications to the norm-optimal ILC algorithm to improve control robustness, ensure smooth control dynamics, and preserve high performance. The proposed method is implemented on a walking beam motion system with a flexibly coupled robotic mechanism. Validation is performed using a nonlinear Simscape/Simulink model that accurately captures multibody dynamics and system nonlinearities.

The model incorporates both feedback and feedforward controllers based on parametric and nonparametric system identification, providing robust control aligned with high-order model behavior. Section “Norm optimal ILC design: Modifications, simulation verification, results, and discussion” presents the ILC algorithm development, simulation setup, and proposed modifications.

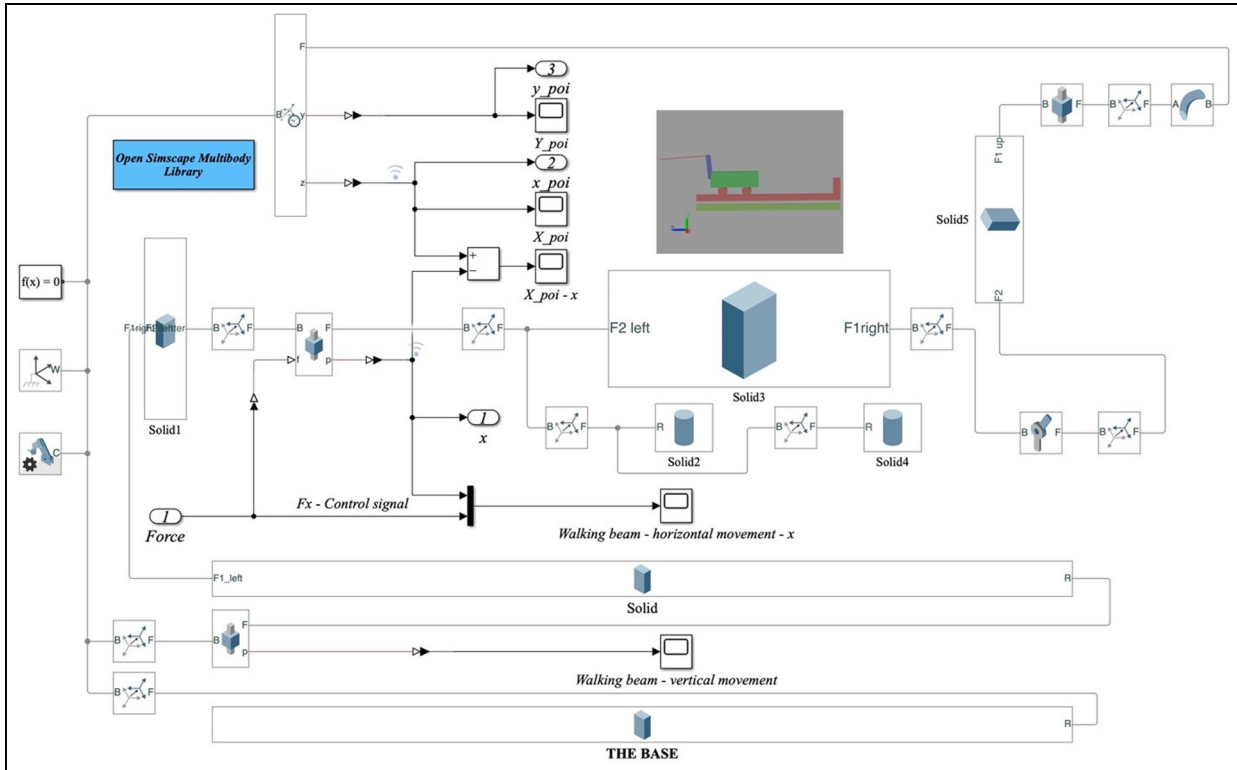
For systems with unstable or poorly damped poles, serial norm-optimal ILC architectures are shown to be more suitable than parallel configurations.<sup>33,34</sup> The proposed approach addresses issues of controller fragility and excessive control input variations,<sup>35–37</sup> enabling stable and efficient system operation. Simulation results in MATLAB/Simulink confirm the effectiveness of the proposed strategy and provide valuable insights into the dynamic behavior of the walking beam motion system.

## System description

To highlight its practical relevance, the walking beam motion system analyzed in this paper reflects typical setups found in automated production lines requiring tightly synchronized transfer and placement operations. In general, such a system with a flexibly coupled robotic mechanism is depicted in Figure 1. The control variable is the force  $F_x$  acting on the walking beam, whose movement is tracked via encoders. The control objective is to ensure that the point of interest  $(x_{poi}, y_{poi})$  follows a specified reference trajectory. This point corresponds to the tip of the gripper, which is used for high-precision assembly tasks within the machine. Although a sensor to track  $(x_{poi}, y_{poi})$  during machine operation is unavailable, it is assumed to be present during the development of the control algorithm that generates the control force  $F_x$ . All parameters shown in the figure are known, enabling the creation of a simulation model of the controlled plant using the Simscape/Simulink module in MATLAB. The gripper is modeled as a thin beam, divided into four finite elements, while

**Table 1.** One example of the possible parameters of the plant from Figure 1.

$c_{\text{base}} = 4\text{e}4 \text{ N/m}$	$m_{\text{stage}} = 1 \text{ kg } (1 \text{ m} \times 0.1 \text{ m} \times 0.40 \text{ m})$	$l_{\text{thin\_beam}} = 1 \text{ m } (0.5 \text{ cm} \times 0.5 \text{ cm})$
$d_{\text{base}} = 100 \text{ N/m/s}$	$c_{\text{low\_arm}} = 200 \text{ Nm/rad}$	$\text{density} = 2700 \text{ kg/m}^3 \text{ \% thin beam}$
$l_{\text{base}} = 3 \text{ m}$	$d_{\text{low\_arm}} = 1 \text{ Nm/(rad/s)}$	$\text{Young\_modulus} = 70 \text{ GPa \% thin beam}$
$M_{\text{base}} = 1000 \text{ kg } (3 \text{ m} \times 0.1 \text{ m} \times 0.15 \text{ m})$	$m_{\text{low\_arm}} = 0.75 \text{ kg } (0.1 \text{ m} \times 0.1 \text{ m} \times 0.6 \text{ m})$	$\text{Poisson\_ratio} = 0.33 \text{ \% thin beam}$
$m_{\text{baseD}} = 10 \text{ kg } (3 \text{ m} \times 0.1 \text{ m} \times 0.15 \text{ m})$	$c_{\text{up\_arm}} = 400 \text{ N/m}$	$N_{\text{finite\_elements}} = 4 \text{ \% thin beam}$
$m_{\text{baseL}} = 1 \text{ kg } (0.15 \text{ m} \times 0.1 \text{ m} \times 0.4 \text{ m})$	$d_{\text{up\_arm}} = 100 \text{ N/(m/s)}$	$M = m_{\text{stage}} + m_{\text{low\_arm}} +$
		$+l_{\text{thin\_beam}}*5\text{e-}3*5\text{e-}3*\text{density}$

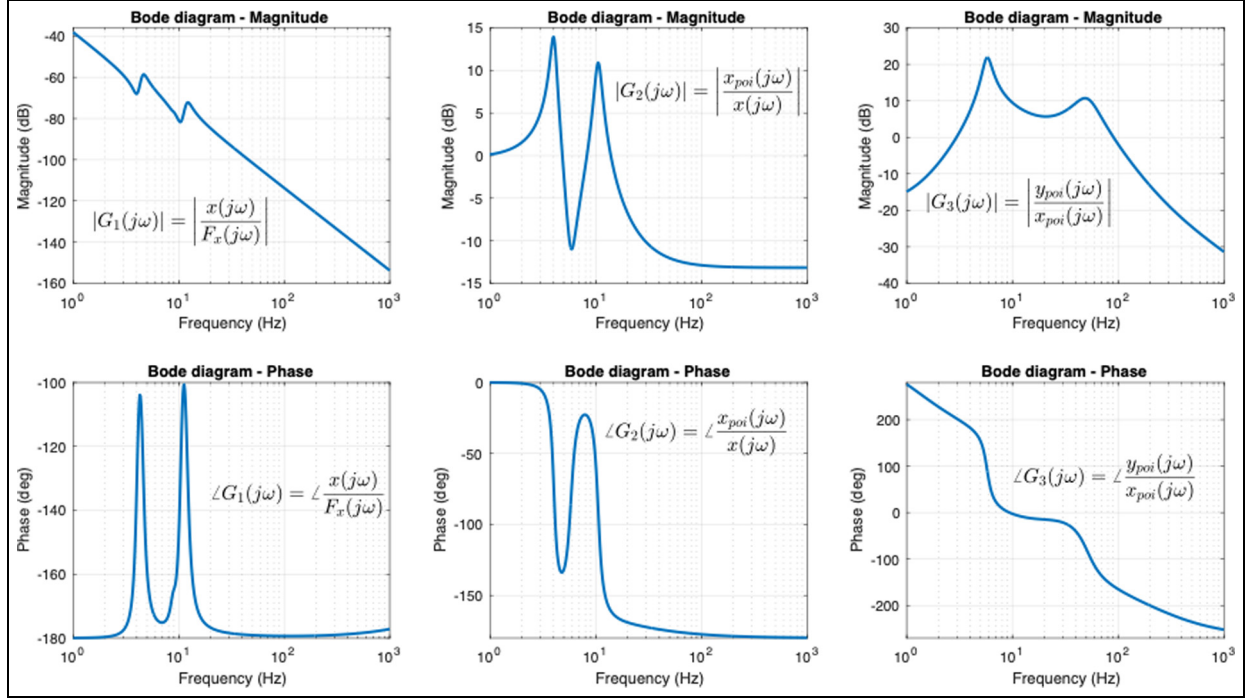
**Figure 2.** Simscape/Simulink model of controlled plant from Figure 1, or walking beam motion system with a flexible coupled robotic mechanism.

the rest of the subsystem is represented by a nonlinear time-invariant model with lumped parameters. Analytical Modeling using Lagrange's equations and linearization of the nonlinear model is also feasible.

Based on the known geometric and physical parameters of the rigid bodies and their connections in the multibody system, a software model can be developed in Simscape/Simulink. Linearization or identification techniques can then be applied to this model to derive suitable models for controller design. The Simscape/Simulink simulation model fully corresponds to the controlled plant (or real experimental setup). Figure 2 presents recorded Bode diagrams, serving as nonparametric models of the controlled plant depicted in Figure 1. The parameters listed in Table 1 correspond to a motion system used in industrial wirebender

machines. While the numerical values are not identical to a real wirebender, the system geometry and the characteristics of the nonparametric model shown in Figure 3—within the desired closed-loop frequency bandwidth—are representative of the actual system.

The Bode diagrams in Figure 3 were generated primarily by estimating frequency response data (FRD models) from simulation-based experiments using broadband noise excitation, including uniform, Gaussian, and PRBS signals. These FRD models formed the core of the analysis and were compared against models obtained through numerical linearization. Coherence functions were computed as well, confirming the reliability of the identification. The resulting FRD models served as the basis for both controller design and performance evaluation.



**Figure 3.** Nonparametric models of the controlled plant shown on Figure 1.

To ensure reliable excitation across a wide frequency band, all input signals were bounded in amplitude to a maximum of 10 N. The experiments were conducted with a sampling time of  $T_s = 2 \times 10^{-4}$  s, corresponding to a sampling frequency of 5000 Hz. The PRBS excitation was generated using a 10-bit shift register, yielding a sequence of  $2^{10} - 1 = 1023$  samples, while white and uniform noise signals provided broadband excitation. In all cases, the usable identification bandwidth ranged approximately from 1 to 400 Hz, which corresponds to roughly twice the expected closed-loop bandwidth of the system.

Walking beam systems are widely deployed in real-world settings where synchronization with adjacent machinery is essential. For example, in metal forming lines, walking beams coordinate with stamping tools and transfer stations to support high-throughput precision manufacturing. In automated packaging, they work in tandem with pick-and-place units to ensure accurate product handling and positioning. These scenarios underscore the practical motivation behind the control strategy developed in this paper.

The structure of the closed-loop system is shown in Figure 4. The feedback loop utilizes the encoder signal to measure the position of the walking beam. The system also includes a feedforward controller,  $C_{ff}(s) = Ms^2$ , where  $M$  represents the total mass driven by the control force  $F_x$  (as detailed in Table 1 and Figure 1). The feedforward controller  $C_{ff}(s) = Ms^2$  operates on the smooth reference trajectory  $x_{ref}(t)$ , which is generated offline, ensuring that noise

amplification is not a concern in this application. The trajectory  $x_{ref}(t)$  used in this study is generated using a quintic polynomial:

$$x_{ref}(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5$$

The coefficients  $a_0$  to  $a_5$  are determined by enforcing boundary conditions on position, velocity, and acceleration at the start ( $t = t_0$ ) and end ( $t = t_f$ ) of the motion interval:

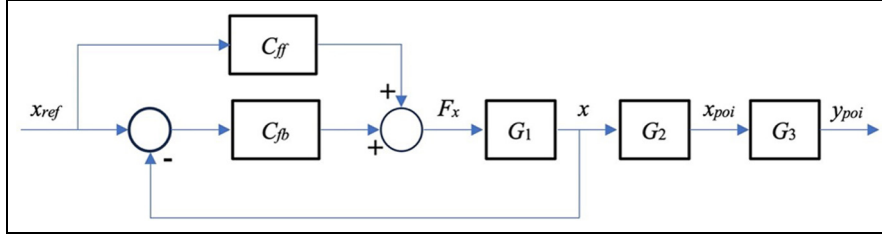
$$x_{ref}(t_0) = q_0, \dot{x}_{ref}(t_0) = \dot{q}_0, \ddot{x}_{ref}(t_0) = \ddot{q}_0$$

$$x_{ref}(t_f) = q_f, \dot{x}_{ref}(t_f) = \dot{q}_f, \ddot{x}_{ref}(t_f) = \ddot{q}_f$$

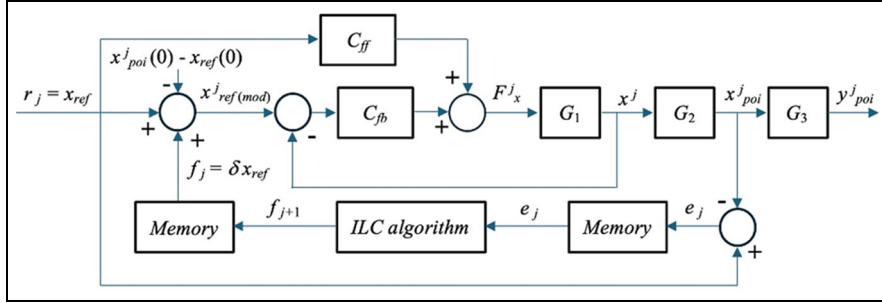
This formulation ensures smooth and continuous reference signals, along with their first and second derivatives. Although the jerk (third derivative) is not explicitly optimized or constrained, it remains bounded and continuous due to the inherent smoothness of the polynomial. The trajectory is typically extended with idle periods before and after the motion to improve transient handling. This trajectory design inherently limits  $\ddot{x}_{ref}$ , thereby reducing dynamic coupling effects and ensuring feasible control effort for both horizontal and vertical motion components.

Consequently, the controller  $C_{ff}(s) = Ms^2$ , shown in Figure 4, can be viewed as equivalent to a controller  $C_{ff}(s) = M$  driven by the available reference acceleration signal  $\ddot{x}_{ref}(t)$ . Thus, noise amplification is not an issue in this context.





**Figure 4.** Structure of the closed-loop walking beam motion system with integrated feedforward controller.



**Figure 5.** Structure of the closed-loop walking beam motion system with flexible coupled robotic mechanism with integrated feedforward controller and serial ILC connection.

The feedback controller,  $C_{fb}$ , is designed in the frequency domain to achieve a closed-loop bandwidth of 215 Hz (refer to the closed-loop system in Figure 4). The design ensures standard stability margins, including a modulus margin below 6 dB, a phase margin above 30°, and a gain margin above 6 dB.<sup>37</sup> The controllers ( $C_{ff}$ ,  $C_{fb}$ ) are implemented digitally in state space with a sampling time of  $T_s = 2 \times 10^{-4}$  s. This sampling time was selected to reflect the timing resolution typically used in industrial high-precision motion systems with fast dynamics and flexible structures,<sup>33,34</sup> such as bonding machines.

However, achieving the desired motion performance for the walking beam's position ( $x$ ) does not necessarily ensure equivalent performance for the point of interest ( $x_{poi}$ ,  $y_{poi}$ ). This point, corresponding to the tip of the gripper, is not rigidly connected to the walking beam motion system. Its motion depends on both the movement of the walking beam and initial conditions, such as the weight of the robotic mechanism's gripper, the resulting bending effects, and the relative motion between the robotic mechanism and the walking beam. Additionally, the dynamic characteristics of the flexible coupling between the robotic mechanism and the walking beam system, as shown in Figure 3, also influence the motion.

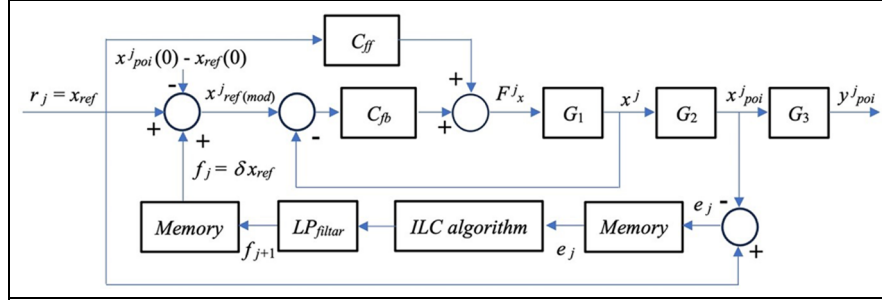
The system's goal is for the point of interest ( $x_{poi}$ ,  $y_{poi}$ ) to follow a specified reference trajectory ( $(x_{ref}, y_{ref})$ ,  $y_{ref} = \text{const.}$ ) which repeats periodically in alignment with defined machine operation cycles ( $j = 1, 2, 3, \dots$ ).

To achieve this, the control structure in Figure 4 can be modified by incorporating an ILC algorithm and its signal  $f_j$ . During the ILC signal learning phase, the point of interest ( $x_{poi}$ ,  $y_{poi}$ ) can be equipped with a sensor (e.g. an accelerometer), as shown in Figure 5. In nominal operation, the ILC controller is replaced by an ILC signal generator tailored to the reference trajectory ( $(x_{ref}, y_{ref})$ ,  $y_{ref} = \text{const.}$ ), and the system operates without a sensor at the point of interest ( $x_{poi}$ ,  $y_{poi}$ ).

The control variable ( $F_x$ ) influences the output position ( $x_{poi}$ ) according to the reference trajectory ( $x_{ref}$ ). The motion of  $y_{poi}$  depends on (1) the initial conditions, (2) the system's physical parameters (e.g. stiffness and damping), and (3) the second derivative of the reference trajectory ( $x_{ref}$ , i.e. reference acceleration).

Improved performance of  $y_{ref} = \text{const.}$  can be achieved by (1) limiting the acceleration of the reference trajectory ( $x_{ref}$ ), (2) increasing the system's damping and stiffness to better support the motion of  $y_{poi}$ , and (3) minimizing the impact of initial conditions on the movement of the point of interest ( $x_{poi}$ ,  $y_{poi}$ ). The following analysis focuses primarily on improving the tracking quality of ( $x_{poi}$ ) with respect to the reference trajectory ( $x_{ref}$ ).

During the development and deployment of the ILC-based control strategy, measurement noise and unmodeled dynamics may influence the learning process. High-frequency noise can be amplified by the ILC signal if not properly addressed. To mitigate such effects and ensure robust performance, low-pass



**Figure 6.** Structure of the closed-loop walking beam motion system with flexible coupled robotic mechanism with integrated feedforward controller and modified serial ILC connection.

filtering of the learned ILC signal is considered, as illustrated in Figure 6. Additionally, the weighting matrices in the ILC design (see Section “Norm optimal ILC design: Modifications, simulation verification, results, and discussion”) are selected to balance learning speed and robustness against disturbances and modeling uncertainties.

The feasibility of filtering major input signals (setpoint, process output, and measurable load disturbances) to the PID controller is analyzed, yielding validated modifications for maintaining good system performance.<sup>36</sup> For instance, a low-pass filter can reduce overshoots during setpoint changes in the setpoint:

$$LP_{filar} = \frac{1}{1 + T_f s}; \quad LP_{filar} = \frac{1}{(1 + T_f s)^2}; \quad (1)$$

The first-order filter eliminates step changes in the proportional component of the controller, while the second-order filter addresses changes in the derivative component during setpoint transitions.<sup>36</sup> In this context, Figure 6 proposes a structural modification (with the final proposed modification presented in Figure 12).

Since  $C_{ff}(s) = Ms^2$ , the second derivative of the reference trajectory significantly influences the dynamics of the control signal. A reference signal with rapid changes may cause the control signal to exceed actuator limits. While reference trajectories are generally designed to avoid unacceptable dynamics in control and output signals, the ILC algorithm aims to precisely track challenging reference trajectories, which may involve rapid dynamic changes. Consequently, the ILC signal can exhibit sharp variations and discontinuities. Similar to the approach in Hägglund,<sup>36</sup> the ILC signal could be filtered in a serial connection to mitigate undesirable dynamics in the control signal. However, proper implementation of the filter is challenging, as it risks adversely affecting previously achieved system performance.

## Norm optimal ILC design: Modifications, simulation verification, results, and discussion

In the previous section, the signals and functionalities of the system components, as well as the overall setup illustrated in Figures 4 to 6, were discussed. This section focuses on the synthesis of the ILC algorithm, which aims to ensure the convergence of the compensation signal  $f_j(t)$  through repeated iterations ( $j = 1, 2, 3, \dots$ ). The goal is to improve the tracking accuracy of the point-of-interest position  $x_{poi}(t)$  along the reference trajectory  $r_j(t) = x_{ref}(t)$ .

The plant is modeled using discrete-time transfer functions  $G_1(z)$  and  $G_2(z)$  (see Figures 3 and 5), while the feedback and feedforward controllers are denoted by  $C_{fb}(z)$  and  $C_{ff}(z)$ , respectively. The Z-transforms  $r_j(z) = x_{ref}(z)$ ,  $f_j(z)$ , and  $x_{poi}^j(z)$  represent the reference trajectory, the ILC compensation signal, and the point-of-interest position at iteration  $j$ , respectively (see Figure 5).

The plant model, including  $G_1(z)$ ,  $G_2(z)$ , and  $G_3(z)$  (see Figures 3 and 5), was identified in the frequency domain using nonparametric Bode plots derived from the nonlinear Simscape/Simulink model, which accurately captures the plant's dynamic behavior under operating conditions.

The complementary sensitivity function of the system without the ILC algorithm ( $f_j(z) = 0$ ) and feedforward controller ( $C_{ff} = 0$ ) in Figure 5 is given by the following expression:

$$T_{cs}(z) = \frac{x_{poi}(z)}{x_{ref}(z)} = \frac{C_{fb}(z)G_1(z)G_2(z)}{1 + C_{fb}(z)G_1(z)} \quad (2)$$

The reference trajectory  $x_{ref}(z)$  is repetitive and remains the same for each repetition cycle  $j$ . Since the reference trajectory is consistent across iterations, the tracking error in the  $j$ -th iteration is given by:

$$E_j(z) = x_{ref}(z) - x_{poi}^j(z) \quad (3)$$

An ILC algorithm can be expressed by its update equation:

$$f_{j+1}(z) = Q(z)(f_j(z) + L(z)E_j(z)) \quad (4)$$

Here,  $L(z)$  is the learning filter, and  $Q(z)$  is the robustness filter. Both filters are implemented offline using non-causal filtering methods. After filtering with  $L(z)$ , time-shifting is applied. The filtering with  $Q(z)$  is designed based on its magnitude response  $|Q(z)|$  only, providing zero-phase filtering. These non-causal filtering methods enable high-precision learning. The learning process is repeated iteratively to reduce the tracking error.

If relation (4) holds, the tracking error propagation for the system in Figure 5 is expressed as:

$$E_{j+1}(z) = Q(z)[1 - T_{cs}(z)L(z)]E_j(z) + [1 - Q(z)]\left[1 - \left(1 + \frac{C_{ff}(z)}{C_{fb}(z)}\right)T_{cs}(z)\right]x_{ref}(z)$$

The first term determines how much of the previous error is propagated to the next iteration. The tracking error converges monotonically if and only if:

$$|Q(z)[1 - T_{cs}(z)L(z)]| < 1 \quad (5)$$

for all frequencies  $f_h$  with  $z = e^{i2\pi f_h T_s}$ . To achieve this, the learning  $L(z)$  filter is ideally chosen to cancel the closed-loop dynamics:

$$L(z) \approx (T_{cs}(z))^{-1} \quad (6)$$

However, this inversion is typically not exact due to system non-idealities or non-minimum phase behavior. In such cases, exact inversion is replaced by stable approximations, such as Zero Phase Error Tracking Control (ZPETC) or regularized inversion, which retain only the stable portion of the inverse. These non-causal approximations are implemented offline to preserve robustness while avoiding unstable dynamics.

Therefore,  $Q(z)$  is introduced as a design parameter to ensure robustness and prevent amplification of noise or modeling errors. The combined use of  $L(z)$  and  $Q(z)$  balances convergence speed and robustness.

For the controlled plant in Figure 1 and the ILC structure with serial connection, calculating  $L(z)$  becomes significantly more complex. The filter  $Q(z)$  is typically a low-pass filter designed to ensure the convergence of the ILC algorithm in (4). For the structure in Figure 5 (with the serially connected ILC algorithm), during iteration  $j$ , consider  $N$  data samples for each relevant signal:

$$\begin{aligned} \mathbf{x}_{poi}^j &= [x_{poi}^j(0) \ x_{poi}^j(1) \ \dots \ x_{poi}^j(N-1)]^T \\ \mathbf{x}_{ref} &= [x_{ref}(0) \ x_{ref}(1) \ \dots \ x_{ref}(N-1)]^T \\ \mathbf{e}_j &= \mathbf{x}_{ref} - \mathbf{x}_{poi}^j = [e_j(0) \ e_j(1) \ \dots \ e_j(N-1)]^T \\ \mathbf{f}_j &= [f_j(0) \ f_j(1) \ \dots \ f_j(N-1)]^T \end{aligned} \quad (7)$$

The impact of ILC signals  $f_j$  on the output variable  $x_{poi}^j$  can be described by the impulse response of the closed-loop system from  $f_j$  to  $x_{poi}^j$ , represented by the impulse response of  $T_{cs}$ :

$$\mathbf{x}_{poi}^j = \mathbf{T}_{cs} \cdot \mathbf{f}_j; \quad \mathbf{T}_{cs} = \begin{bmatrix} T_{cs}(0) & 0 \\ \vdots & \ddots \\ T_{cs}(N-1) & T_{cs}(0) \end{bmatrix}; \quad (8)$$

To derive the optimal update law for the ILC compensation signal  $f_{j+1}$ , we begin with the expression of the tracking error at iteration  $j+1$  in terms of the previous error  $e_j$  and the change in compensation signal:

$$\mathbf{e}_{j+1} = \mathbf{e}_j - \mathbf{T}_{cs}(\mathbf{f}_{j+1} - \mathbf{f}_j) \quad (9)$$

This relation assumes that the closed-loop system from  $f_j$  to  $x_{poi}^j$  is linear and time-invariant (LTI), and that the same reference trajectory  $x_{ref}$  is repeated across iterations, as is standard in ILC frameworks.

We now define a quadratic cost function that penalizes the tracking error  $\mathbf{e}_{j+1}$ , the control effort  $\mathbf{f}_{j+1}$ , and the variation  $\mathbf{f}_{j+1} - \mathbf{f}_j$ :

$$\mathcal{J} = \mathbf{e}_{j+1}^T \mathbf{W}_e \mathbf{e}_{j+1} + \mathbf{f}_{j+1}^T \mathbf{W}_f \mathbf{f}_{j+1} + (\mathbf{f}_{j+1} - \mathbf{f}_j)^T \mathbf{W}_{\Delta f} (\mathbf{f}_{j+1} - \mathbf{f}_j) \quad (10)$$

Substituting the expression for  $\mathbf{e}_{j+1}$ , we obtain:

$$\begin{aligned} \mathcal{J} &= [\mathbf{e}_j - \mathbf{T}_{cs}(\mathbf{f}_{j+1} - \mathbf{f}_j)]^T \mathbf{W}_e [\mathbf{e}_j - \mathbf{T}_{cs}(\mathbf{f}_{j+1} - \mathbf{f}_j)] \\ &\quad + \mathbf{f}_{j+1}^T \mathbf{W}_f \mathbf{f}_{j+1} + (\mathbf{f}_{j+1} - \mathbf{f}_j)^T \mathbf{W}_{\Delta f} (\mathbf{f}_{j+1} - \mathbf{f}_j) \end{aligned} \quad (11)$$

Expanding this term:

$$\begin{aligned} \mathcal{J} &= \mathbf{e}_j^T \mathbf{W}_e \mathbf{e}_j - 2(\mathbf{f}_{j+1} - \mathbf{f}_j)^T \mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{e}_j \\ &\quad + (\mathbf{f}_{j+1} - \mathbf{f}_j)^T \mathbf{T}_{cs}^T \cdot \mathbf{W}_e \mathbf{T}_{cs} (\mathbf{f}_{j+1} - \mathbf{f}_j) \\ &\quad + \mathbf{f}_{j+1}^T \mathbf{W}_f \mathbf{f}_{j+1} + (\mathbf{f}_{j+1} - \mathbf{f}_j)^T \mathbf{W}_{\Delta f} (\mathbf{f}_{j+1} - \mathbf{f}_j) \end{aligned}$$

The goal of the norm-optimal ILC design is to determine the compensation control signal  $\mathbf{f}_{j+1}$  in iteration  $j+1$  that minimizes the quadratic cost function  $\mathcal{J}$ , leading to stationary condition:



$$\frac{\partial \mathcal{J}}{\partial \mathbf{f}_{j+1}} = 0$$

Differentiating the cost function with respect to  $\mathbf{f}_{j+1}$ , we obtain:

$$\begin{aligned} \frac{\partial \mathcal{J}}{\partial \mathbf{f}_{j+1}} = & -2\mathbf{T}_{cs}^T \mathbf{W}_e [\mathbf{e}_j - \mathbf{T}_{cs} \cdot \mathbf{f}_{j+1} + \mathbf{T}_{cs} \cdot \mathbf{f}_j] \\ & + 2\mathbf{W}_f \mathbf{f}_{j+1} + 2\mathbf{W}_{\Delta f} (\mathbf{f}_{j+1} - \mathbf{f}_j) = 0 \end{aligned}$$

Rearranging the terms gives the linear system:

$$\begin{aligned} (\mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{T}_{cs} + \mathbf{W}_f + \mathbf{W}_{\Delta f}) \mathbf{f}_{j+1} = & \mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{e}_j \\ & + (\mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{T}_{cs} + \mathbf{W}_{\Delta f}) \mathbf{f}_j \end{aligned}$$

Solving the update equation yields the ILC update law:

$$\mathbf{f}_{j+1} = \mathbf{L}_o \cdot \mathbf{e}_j + \mathbf{Q}_o \cdot \mathbf{f}_j \quad (12)$$

Here, the learning and robustness matrices are defined as:

$$\mathbf{L}_o = [\mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{T}_{cs} + \mathbf{W}_f + \mathbf{W}_{\Delta f}]^{-1} \mathbf{T}_{cs}^T \cdot \mathbf{W}_e \quad (13)$$

$$\mathbf{Q}_o = [\mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{T}_{cs} + \mathbf{W}_f + \mathbf{W}_{\Delta f}]^{-1} [\mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{T}_{cs} + \mathbf{W}_{\Delta f}] \quad (14)$$

These expressions clearly demonstrate how the learning behavior is influenced by both system dynamics and the design of the weighting matrices. To ensure numerical stability and robustness of the ILC update law (12), we note that the matrix to be inverted in equations (13) and (14),

$$\mathbf{A} = \mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{T}_{cs} + \mathbf{W}_f + \mathbf{W}_{\Delta f}$$

is symmetric and positive definite, provided that  $\mathbf{W}_f$  and  $\mathbf{W}_{\Delta f}$  are positive semidefinite. This guarantees the existence of a well-defined inverse. The conditioning of  $\mathbf{A}$  can be controlled through the choice of weighting matrices and directly affects the sensitivity of the computed  $\mathbf{L}_o$  and  $\mathbf{Q}_o$  to numerical errors. For the selected weights, the condition number of  $\mathbf{A}$  remains moderate (on the order of  $10^5$ ), which is acceptable for double-precision computation. If needed, conditioning can be further improved by increasing  $\mathbf{W}_f$  or introducing a small positive  $\mathbf{W}_{\Delta f}$ , as these terms act as regularization components. These properties ensure reliable matrix inversion and contribute to the numerical stability of the ILC iteration across a wide range of reference signals and operating conditions.

The matrices  $\mathbf{L}_o$  and  $\mathbf{Q}_o$  are computed once based on the nominal system dynamics and weighting matrices, and remain constant throughout the ILC iterations. This design choice ensures computational efficiency, consistency, and numerical stability.

Generalization is achieved by appropriately selecting the weighting matrices to capture the expected variations in system dynamics and disturbances. As a result, the fixed matrices  $\mathbf{L}_o$  and  $\mathbf{Q}_o$  provide satisfactory performance across a range of operating conditions, without requiring re-computation at each iteration.

To analyze the convergence of the update law (12) and the tracking error signal in the closed-loop control structure shown in Figure 5, we first express the tracking error as:

$$\mathbf{e}_j = \mathbf{x}_{ref} - \mathbf{T}_{cs} \cdot (\mathbf{f}_j + \mathbf{x}_{ref}) = (\mathbf{I} - \mathbf{T}_{cs}) \cdot \mathbf{x}_{ref} - \mathbf{T}_{cs} \cdot \mathbf{f}_j$$

Substituting this into the update law (12) yields:

$$\mathbf{f}_{j+1} = \mathbf{L}_o \cdot (\mathbf{I} - \mathbf{T}_{cs}) \cdot \mathbf{x}_{ref} + (\mathbf{Q}_o - \mathbf{L}_o \cdot \mathbf{T}_{cs}) \cdot \mathbf{f}_j$$

This expression reveals that the evolution of the ILC signal across iterations is governed by the iteration matrix:

$$\mathbf{A} = \mathbf{Q}_o - \mathbf{L}_o \cdot \mathbf{T}_{cs}.$$

Thus, a sufficient condition for the asymptotic convergence of the ILC signal to a steady-state solution is that the spectral norm of  $\mathbf{A}$  is strictly less than one:

$$\|\mathbf{A}\|_{i2} = \|\mathbf{Q}_o - \mathbf{L}_o \cdot \mathbf{T}_{cs}\|_{i2} < 1 \quad (15)$$

where  $\|\cdot\|_{i2}$  denotes the induced 2-norm, equal to the largest singular value of a given matrix. In MATLAB, the singular value decomposition (SVD) function can be used to compute this norm, providing a robust and accurate estimate of the induced 2-norm.

Moreover, by substituting the ILC update equation (12) into the tracking error dynamics equation (9), we obtain:

$$\mathbf{e}_{j+1} = (\mathbf{I} - \mathbf{T}_{cs} \mathbf{L}_o) \mathbf{e}_j - \mathbf{T}_{cs} (\mathbf{Q}_o - \mathbf{I}) \cdot \mathbf{f}_j$$

This equation shows that the tracking error dynamics depend not only on the current error but also on the asymptotic behavior of the ILC signal  $\mathbf{f}_j$ . However, the dominant factor determining error convergence is the matrix  $\mathbf{I} - \mathbf{T}_{cs} \mathbf{L}_o$ . Therefore, a sufficient condition for the convergence of the tracking error is:

$$\|\mathbf{I} - \mathbf{T}_{cs} \mathbf{L}_o\|_{i2} < 1 \quad (16)$$

Notably, the inhomogeneous term

$$\mathbf{Q}_o - \mathbf{L}_o \cdot \mathbf{T}_{cs} = [\mathbf{T}_{cs}^T \cdot \mathbf{W}_e \cdot \mathbf{T}_{cs} + \mathbf{W}_f + \mathbf{W}_{\Delta f}]^{-1} \mathbf{W}_{\Delta f}$$

indicates that tuning  $\mathbf{W}_{\Delta f}$  can directly influence the iterative dynamics and robustness properties.

The convergence analysis highlights that the proposed norm-optimal ILC framework offers a structured approach for shaping the learning dynamics via the design of weighting matrices  $\mathbf{W}_e$ ,  $\mathbf{W}_f$ , and  $\mathbf{W}_{\Delta f}$ .

The matrix  $\mathbf{L}_o$  determines how aggressively the algorithm adapts to the current tracking error, while  $\mathbf{Q}_o$

reflects the memory of the previous ILC signal. The term  $\mathbf{Q}_o - \mathbf{L}_o \cdot \mathbf{T}_{cs}$  plays a key role in the stability of the iteration. By appropriately tuning  $\mathbf{W}_f$  and  $\mathbf{W}_{\Delta f}$ , one can regulate its magnitude and enforce convergence. Specifically:

- (1) A larger  $\mathbf{W}_f$  penalizes large control updates, promoting monotonic convergence,
- (2) The matrix  $\mathbf{W}_{\Delta f}$  acts as a damping term, limiting inter-iteration variation and enhancing robustness to noise and modeling uncertainties. A small  $\mathbf{W}_{\Delta f}$  (i.e. lower penalty on input variation) results in faster convergence, but may reduce robustness to modeling errors or disturbances. Conversely, a larger  $\mathbf{W}_{\Delta f}$  enhances robustness but slows down learning. This highlights a critical trade-off in tuning  $\mathbf{W}_f$  and  $\mathbf{W}_{\Delta f}$  to achieve the desired balance between learning speed and noise sensitivity.

Through proper tuning, the designer can balance convergence speed, robustness, and steady-state performance. The norm condition in equation (15) offers a practical criterion for verifying convergence and guiding the selection of weighting matrices.

Tuning of the weighting matrices<sup>38</sup>:

- (1)  $\mathbf{W}_e$ : Set to reflect the desired weighting of the tracking error. It determines the relative importance of minimizing the error and should be selected based on the application's performance objectives.
- (2)  $\mathbf{W}_f$ : Start with  $\|\mathbf{W}_f\|_{i2}$  significantly larger than  $\|\mathbf{T}_{cs}\|_{i2}$ . A larger  $\mathbf{W}_f$  penalizes excessive control effort, promoting monotonic convergence. Gradually decrease  $\mathbf{W}_f$  until the system begins to diverge, then fine-tune it to achieve an optimal balance between convergence speed and stability.
- (3)  $\mathbf{W}_{\Delta f}$ : Initialize with  $\|\mathbf{W}_{\Delta f}\|_{i2} = 0$  and progressively increase it until steady-state error fluctuations fall within acceptable bounds or stop improving. This term acts as a damping factor, reducing oscillations between iterations and improving robustness to noise and model uncertainty.

In the control system defined by Figures 1 and 5 (or 6), with controllers  $C_{ff}$  and  $C_{fb}$  designed as described in Section "System description," and with the reference trajectory (7) defined over a finite time horizon of  $N$  samples, and parameters listed in Table 1, the following weighting matrices were used in the norm-optimal ILC cost function (10):

$$\mathbf{W}_e = \mathbf{I}, \mathbf{W}_f = 10^{-5} \cdot \mathbf{I}, \mathbf{W}_{\Delta f} = 0 \cdot \mathbf{I}$$

where  $\mathbf{I}$  denotes the identity matrix of size  $N \times N$ , corresponding to the length of the reference trajectory. This configuration was selected based on a widely adopted heuristic tuning approach,<sup>38</sup> which has been validated both in the literature and in practical applications.

In Figure 6, we adopt a third-order Butterworth digital low-pass filter  $LP_{filter}$  with a bandwidth of  $0.999 \cdot (1/T_s)$ , where the sampling time is  $T_s = 2e-4$  s. The selected cutoff frequency (99.9% of maximum) ensures identical structures between Figures 5 and 6. However, simulations show that even minimal controller adjustments significantly impact system performance. This phenomenon, which affects both stability and performance, is referred to as controller fragility.<sup>35</sup>

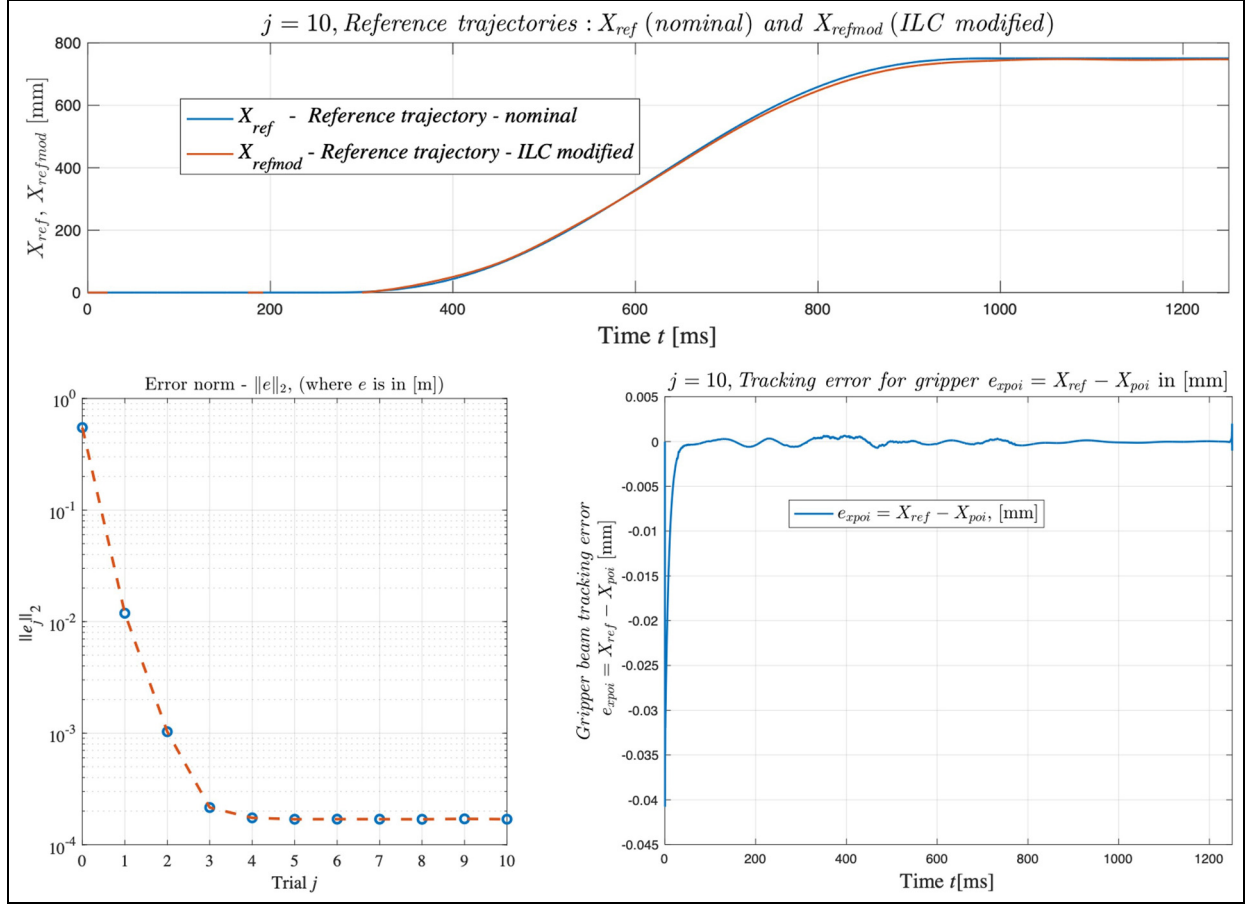
Simulation results for the structure in Figure 6 and the norm-optimal ILC design procedure are shown in Figures 9 and 10. The presence of the  $LP_{filter}$  in Figure 6 causes pronounced oscillations in the control variable. In contrast, the structure in Figure 5 shows the control variable peaking at approximately  $1.5e+4$  N at the beginning and end of the simulation, with no noise in the control signal (Figures 7, 8, 10 and 11).

The tracking error is reduced by over 1000 times with the application of the norm-optimal ILC algorithm. This algorithm enables more efficient tracking of the point of interest along the  $x$ -coordinate ( $x_{poi}$ ). However, since the movement of the point of interest along the  $y$ -coordinate ( $y_{poi}$ ) depends on the second derivative of the reference trajectory, system parameters, and initial conditions. As a result, the ILC algorithm has minimal impact on  $y_{poi}$  movement.

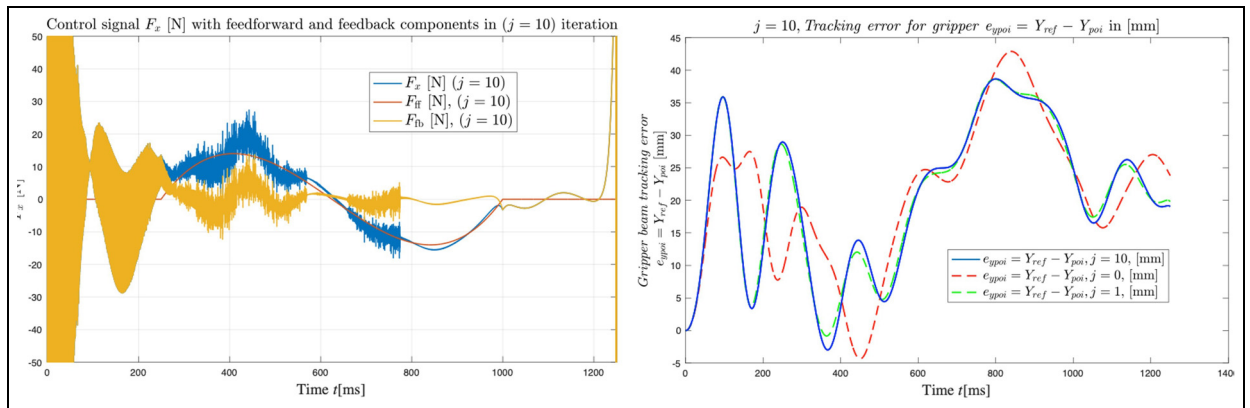
Simulation results indicate that while the norm-optimal ILC algorithm significantly improves the performance of the output variable, it is impractical due to its tendencies toward controller fragility, abrupt jumps, oscillations, and potentially excessive control variable values.

### Modified norm optimal ILC

The ILC signal  $f_j$  is calculated offline between consecutive iterations, aiming to achieve the most efficient tracking of the reference trajectory  $x_{ref}$ . In offline mode, non-causal filtering of the ILC signal  $f_j$  is feasible, allowing noise suppression and reduction of abrupt changes in the control signal. However, such filtering can negatively impact the previously achieved performance of the output signal. Despite the small sampling time ( $T_s = 2e-04$  s), even a delay or lead in the filtered ILC signal as small as one sampling interval can significantly affect output signal performance.



**Figure 7.** Illustration of the iterative convergence of the norm-optimal ILC algorithm and its impact on reducing the tracking error of the x-coordinate of the point of interest.

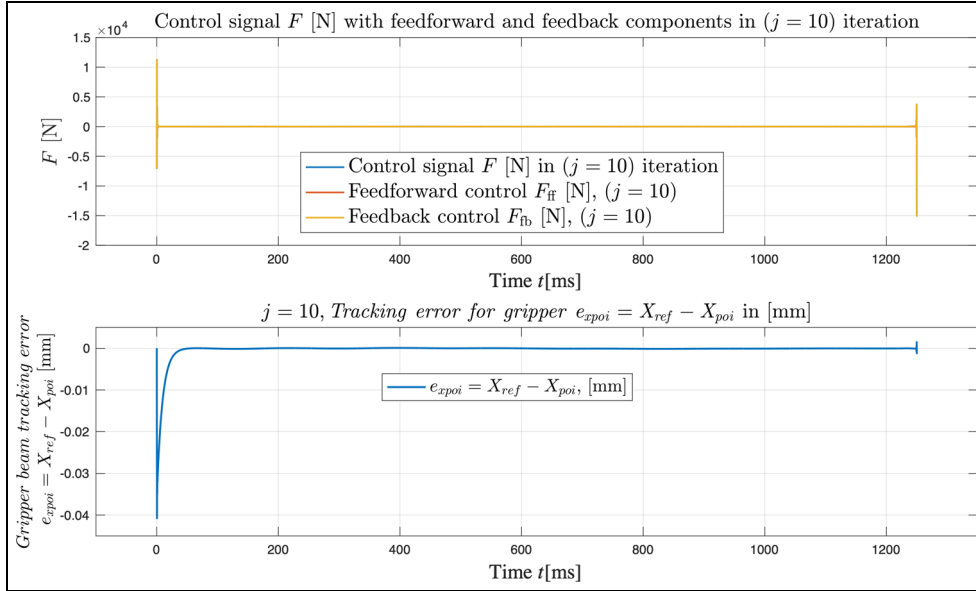


**Figure 8.** Illustration of the operation of the designed norm-optimal ILC structure from Figure 6.

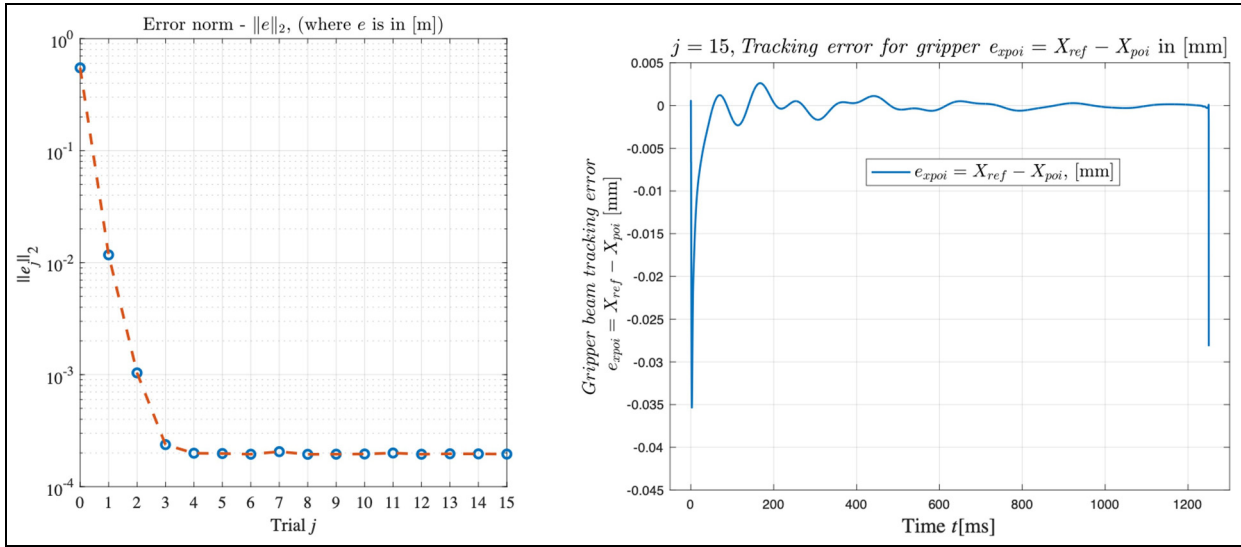
To address controller fragility, we propose a heuristic solution validated through simulations. Specifically, we recommend incorporating a digital non-causal FIR filter:

$$LP_{filter}(z^{-1}) = \frac{z^{-1} + 1 + z}{3} \quad (17)$$

into the structure shown in Figure 6. This filter supplements the third-order Butterworth digital low-pass filter, previously applied with a bandwidth of  $0.999 \cdot (1/T_s)$ , which caused controller fragility (as demonstrated in Figure 10). Simulation results for this modification are presented in Figures 12 and 13 and should be compared with Figures 9 and 10. This approach provides



**Figure 9.** Illustration of the operation of the designed norm-optimal ILC structure from Figure 5.



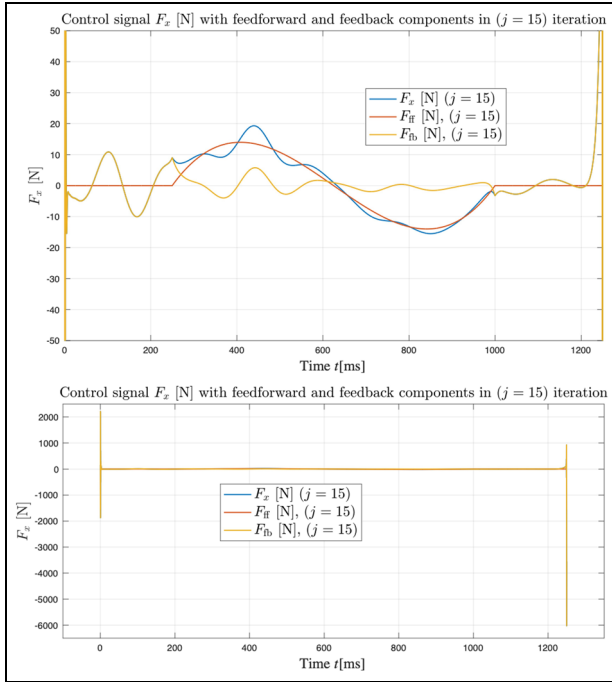
**Figure 10.** Illustration of the iterative convergence of modified norm-optimal ILC algorithm and its impact on reducing the tracking error of the x-coordinate of the point of interest.

acceptable control variable dynamics with minimal degradation in tracking the reference signal  $x_{ref}$ . However, it does not entirely eliminate the tendency of the control variable to exhibit singularities at the first and last samples. To reduce artifacts in the ILC signal near the trajectory boundaries, two general strategies are applied. First, the reference trajectory is extended by a short time interval (e.g. 50 ms) beyond the operational time window within which the ILC signal is implemented. This extension allows smoother convergence of the learned ILC signal near the endpoint.

Second, the first and last samples of the control signal are smoothed by aligning them with adjacent values (e.g. setting the last sample equal to the penultimate one), thereby preventing discontinuities or numerical peaks. Between iterations, non-causal filtering is performed offline, which can also compensate for time delays in the implementation of the ILC signal  $f_j$ . This general method improves signal continuity, especially in the presence of non-causal filtering, and is applicable to a wide range of ILC implementations. A more general recommendation for the structure in Figure 6 is as

follows: The filtered control signal's averaged sample is calculated by averaging  $n_- + 1 + n_+$  samples of the original signal, where the number of samples ahead ( $n_+$ ) and behind ( $n_-$ ) the current sample are equal ( $n_- = n_+$ ). The filtered ILC signal is then subjected to non-causal time-shifting by ( $n_+$ ) samples forward. To balance filtering performance and suppression of abrupt jumps or flickering in the control signal,  $n_+$  should be as small as possible ( $n_+ = 1$ ).

The FIR filter defined by (17) is a symmetric 3-point moving average filter, with frequency response:  $LP_{filter}(e^{j\omega}) = \frac{1}{3}(1 + 2 \cos \omega)$



**Figure 11.** Control signal of the modified norm-optimal ILC structure from Figure 6. (A non-causal low-pass filter (17) is used, and the last sample of the ILC signal is set to be equal to the penultimate one.)

where  $\omega$  is the normalized angular frequency (rad/sample). The filter has unity gain at  $\omega = 0$  and decreasing gain toward the Nyquist frequency ( $\omega = \pi$ ), smoothing the control signal by suppressing high-frequency noise and abrupt changes. Being symmetric and non-causal, it causes zero phase distortion and introduces a one-sample forward time shift, consistent with its non-causal operation. This corresponds to a time delay of exactly one sampling interval, that is,  $1T_s = 0.0002$  s (0.2 ms). The frequency domain effects of this filter are shown in Figure 13. Overall, this FIR filter effectively balances noise reduction and delay, improving control performance and signal smoothness.

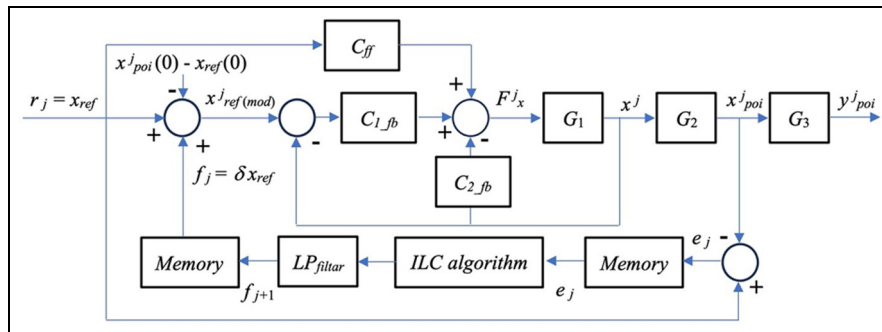
The modification in Figure 12 introduces the feedback controller implementation as follows:

$$C_{fb}(z) = C_{1fb}(z) + C_{2fb}(z); C_{1fb}(z) = \frac{k_1 z}{z - 1} = \frac{k_1}{1 - z^{-1}}; \quad (18)$$

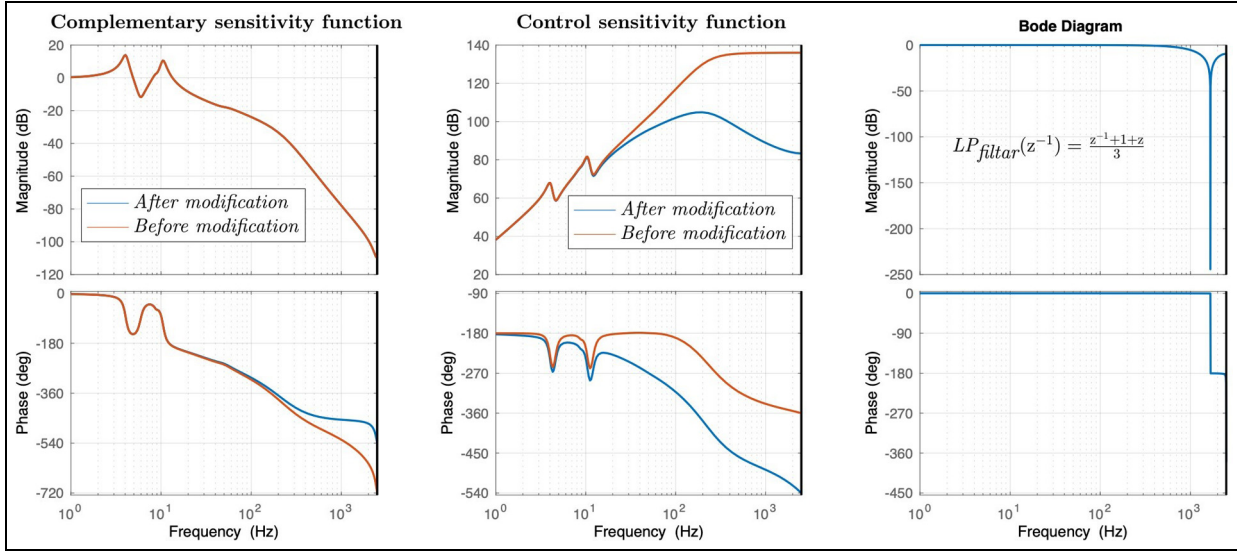
The integral action ( $C_{1fb}$ ) of the feedback controller ( $C_{fb}$ ) remains in the direct branch, while the rest of the controller ( $C_{2fb}$ ) is moved to the feedback branch. This modification does not affect the system's characteristic equation, sensitivity function, or load sensitivity function. Stability parameters—such as a modulus margin below 6 dB (4.7 dB), a phase margin above  $30^\circ$  ( $36.8^\circ$ ), and a gain margin above 6 dB (15.8 dB)—remain unchanged.

Based on the results in Figure 13, we conclude that the filter in equation (17) is unnecessary in the structure presented in Figure 12.

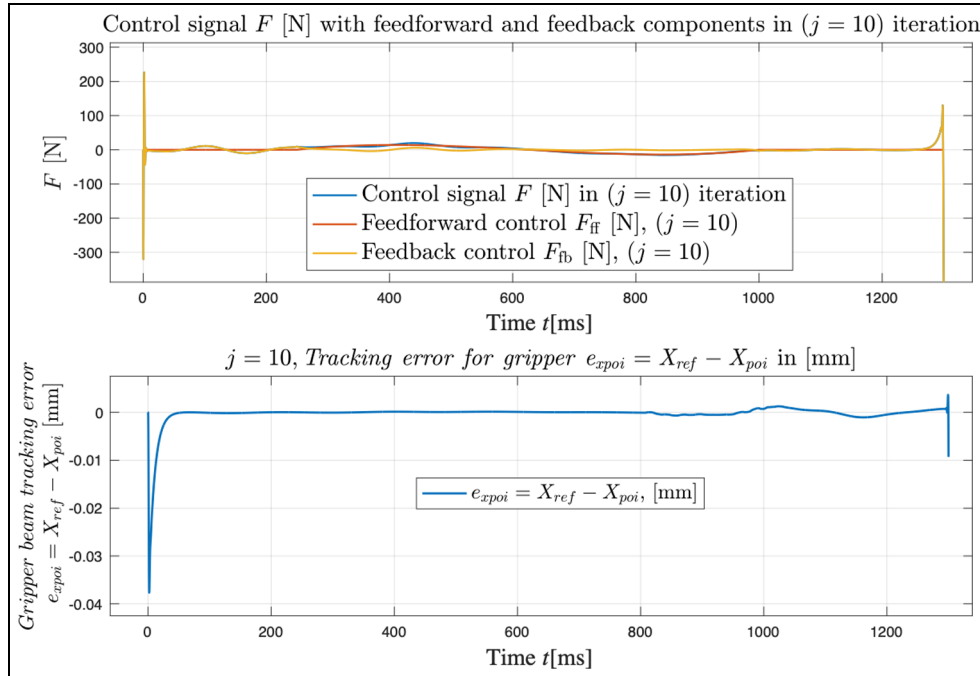
Excessive control actions observed at the end of the reference trajectory (Figure 9) are attributed to numerical instability. Extending the reference trajectory by a suitable time interval,  $\Delta t$  (e.g. 50 ms), enables the offline computation of the ILC signal  $f_j$  to continue beyond the actual system implementation. This approach prevents abrupt jumps in the control signal at the trajectory endpoint.



**Figure 12.** Structure of the closed-loop walking beam motion system with flexible coupled robotic mechanism and modified serial ILC connection and feedback controller implementation.



**Figure 13.** Complementary sensitivity function ( $T_{cs} = x_{poi}/x_{ref}$ )—before (2) and after modification (18) on Figure 12 Control sensitivity function ( $CS = F_x/x_{ref}$ )—before and after modification (18) on Figure 12; and filter (17) on Figure 12.



**Figure 14.** Illustration of the operation of the designed norm-optimal ILC structure from Figure 12.

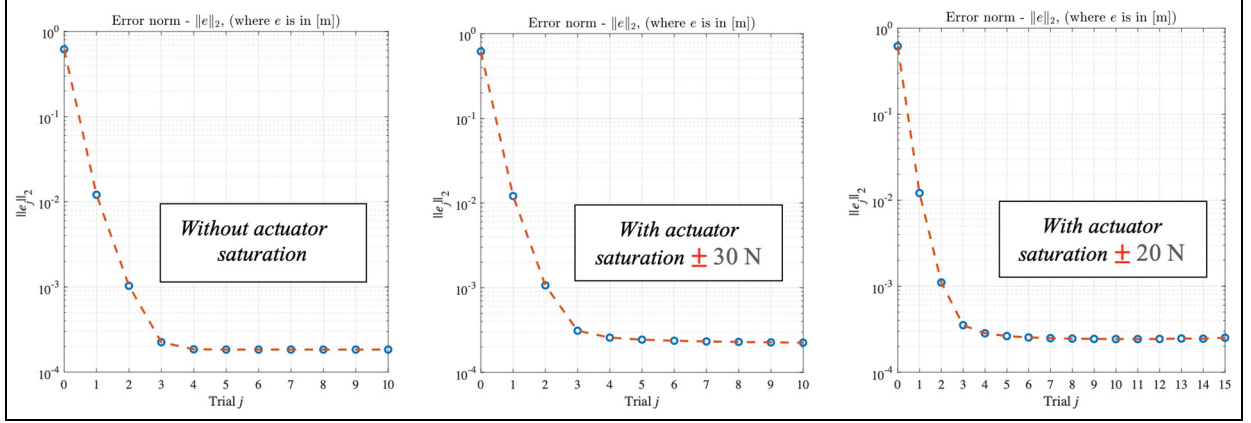
At the beginning of the reference trajectory, sudden variations in the ILC signal—often caused by disturbances such as initial conditions—can also lead to excessive control actions and actuator saturation. The modified structure in Figure 12 effectively mitigates these discontinuities in the control variable.

This improvement becomes evident when comparing the control signal amplitudes across different

configurations: approximately 10,000 N in Figure 9, 2000 N in Figure 11, and only 300 N in Figure 14. Considering realistic actuator limitations, the control signal in Figure 14 remains well within the acceptable range, in contrast to the excessive peaks observed in Figures 9 and 11.

The effectiveness of the proposed modification becomes particularly evident under stringent actuator





**Figure 15.** Illustration of the iterative convergence process for the tracking error norm in the structural implementation of the system from Figure 12, under different control signal constraints.

saturation limits. Figure 15 explicitly illustrates the iterative convergence of the tracking error norm under different actuator saturation thresholds, confirming that the modified ILC structure maintains system performance even when the actuator saturation level is as low as 30 or 20 N.

The fragility observed in our study does not stem from internal closed-loop instability, but rather from signal-level degradation in the control signal. These degradations are triggered by abrupt changes in the feedback controller input, caused by reference trajectory endpoints, discontinuities in the ILC signal, or mismatches in initial conditions. This phenomenon is particularly relevant for real-world implementations, where actuator saturation and excessive oscillations must be avoided.

In this paper, we characterize controller fragility as the sensitivity of control signal quality to small, abrupt perturbations in initial conditions, the reference trajectory, the ILC signal and other internal signal discontinuities, measurement noise, and minor variations in the controller's structure and parameters. Specifically, we define controller fragility in terms of undesirable increases in control signal amplitude, sudden jumps, or high-frequency oscillations, triggered by minor structural or signal-level changes.

To quantify this phenomenon, we define the Control Signal Fragility Index (CSFI) as:

$$CSFI \pm = \frac{\|F_{x, \pm \Delta}\|_2 - \|F_{x, nominal}\|_2}{\|F_{x, nominal}\|_2} \quad (19)$$

where  $\|F_x\|_2$  denotes the L2-norm of the control signal  $F_x$ , and  $\Delta$  represents a small impulse perturbation in the controller input or a structured variation, such as a  $\pm 1$ -sample delay in the ILC signal, a minor gain change in a controller parameter, or a slight structural modification.

CSFI value greater than 0.5 indicates significant controller fragility—that is, a high sensitivity of the control signal to small, abrupt internal signal variations. The simulation results presented in Figures 8, 9, 11, and 14 suggest that unmodified controller structures exhibit high CSFI values, confirming their vulnerability. In contrast, the proposed modifications (e.g. the structure shown in Figure 12) result in markedly reduced fragility, consistent with lower CSFI values and improved robustness.

This paper introduces a general methodology aimed at reducing the sensitivity of the control variable to abrupt variations in the ILC signal when the signal is applied in series within a closed-loop control architecture. In this way, in serial ILC implementations within a closed-loop control structure, abrupt variations in the ILC signal can negatively affect both horizontal and vertical motion components. The proposed method relies on a structural decomposition of the feedback controller, whereby the integral component is retained in the direct path, and the remaining dynamics are relocated to the feedback path. This modification effectively mitigates, and in some cases eliminates, control signal degradation caused by discontinuities in the ILC signal.

Such discontinuities often arise because of attempts to compensate for disturbances associated with initial conditions. Consequently, when feasible, it is preferable to eliminate these disturbances through constructive (i.e. system-level or initialization) measures rather than through compensatory control action.

## Conclusion

The walking beam motion system with a flexibly coupled robotic mechanism is widely used in industrial applications such as assembly lines, high-precision machines, and transport systems. Operating in

repetitive cycles, it tracks a reference trajectory with high precision. However, increasing machine speeds to boost productivity poses challenges, as demanding reference trajectories can lead to more aggressive control signal changes and increased tracking errors. To maintain high-precision tracking of the desired trajectory, an exogenous or feedforward ILC signal can be integrated into the system.

This paper demonstrates that while the ILC signal can enhance precision, its integration into an existing feedback control system may introduce discontinuities and fluctuations in the control signal, potentially degrading overall performance. To address these challenges, the paper proposes structural modifications to the feedback control algorithm and the implementation of the ILC signal, ensuring improved system robustness and precision.

The paper begins with a literature review, highlighting that ILC implementations are highly specific to applications and that the associated implementation challenges represent an area for future research. It then presents a detailed procedure for synthesizing and implementing the norm-optimal ILC signal in the walking beam motion system. The findings show that, even when stability is positively confirmed, inadequate implementation of the feedback controller or improper integration of the ILC signal can result in unacceptable system performance and controller fragility.

Through simulation, the paper proposes and validates a robust implementation strategy for the feedback controller and ILC signal integration. All nonlinear effects in the walking beam motion system are accounted for in the simulation, and the norm-optimal approach to ILC signal synthesis is data-driven rather than model-based.

The proposed approach successfully mitigates controller fragility induced by the ILC signal, ensures feasible control variable dynamics, and achieves the desired system performance.

Although nonlinear friction was not explicitly modeled in the simulation environment, in practical implementation scenarios, friction compensation should be considered. If friction effects are consistent across repetitions, the ILC algorithm is expected to implicitly compensate for them. Otherwise, specialized control strategies for friction compensation may be used in combination with ILC.

Future work will focus on experimental validation of the proposed ILC methodology. A wirebonding machine is considered as a testbed for experimental validation of the proposed ILC methodology under real-world constraints.

## Funding

The authors received no financial support for the research, authorship, and/or publication of this article.


## Declaration of conflicting interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

## References

1. Zhang Y, Chu B and Shu Z. Parameter optimal iterative learning control design: from model-based, data-driven to reinforcement learning. *IFAC Pap Online* 2022; 55(12): 494–499.
2. Wang Y, Gao F and Doyle FJ. Survey on iterative learning control, repetitive control, and run-to-run control. *J Process Control* 2009; 19(10): 1589–1600.
3. Bristow DA, Tharayil M and Alleyne AG. A survey of iterative learning control: a learning-based method for high-performance tracking control. *IEEE Control Syst Mag* 2006; 26(3): 96–114.
4. Ahn HS, Chen Y and Moore KL. Iterative learning control: brief survey and categorization. *IEEE Trans Syst Man Cybern C Appl Rev* 2007; 37(6): 1099–1121.
5. Matijevic M and Kostic D. Iterative learning control—an overview of theoretical approaches and industrial applications. In: *Proceedings of Annual Conference of the Scientific Society of Serbia, GODKON NDS 2024*, Kolarac Foundation, 31-October 1-November 2024, Belgrade, Serbia.
6. Garden M. Learning control of actuators in control systems. US Patent 3555252, 1971.
7. Uchiyama M. Formation of high-speed motion pattern of a mechanical arm by trial. *Trans Soc Instrum Control Eng* 1978; 14(6): 706–712.
8. Arimoto S, Kawamura F and Miyazaki S. Bettering operation of robots by learning. *J Robot Syst* 1984; 1(2): 123–140.
9. Moore KL. *Iterative learning control for deterministic systems*. Springer-Verlag, 1993.
10. Rogers E, Chu B, Freeman C, et al. *Iterative learning control algorithms and experimental benchmarking*. Wiley, 2023.
11. Shen D and Wang Y. Survey on stochastic iterative learning control. *J Process Control* 2014; 24(12): 64–77.
12. Shen D. Iterative learning control with incomplete information: a survey. *IEEE CAA J Autom Sin* 2018; 5: 885–901.
13. Matijević MS, Filipović V and Kostić D. Iterative learning control in manufacturing systems: design of ILC algorithms and overview of model inversion techniques for ILC synthesis. In: *International congress motor vehicles & motors*, Kragujevac, Serbia, 10–11 October 2024.
14. Strijbosch N and Oomen TA. Iterative learning control for intermittently sampled data: monotonic convergence, design, and applications. *Automatica* 2022; 139: 110171.
15. Oomen T. Advanced motion control for precision mechatronics: control, identification, and learning of complex systems. *IEEE J Ind Appl* 2018; 7(2): 127–140.

## ORCID iD

Saša Ćuković  <https://orcid.org/0000-0003-0525-2197>

16. Goubej M, Meeusen S, Mooren N, et al. Iterative learning control in high-performance motion systems: from theory to implementation. In: *2019 24th IEEE international conference on emerging technologies and factory automation (ETFA)*, Zaragoza, Spain, 10–13 September 2019, pp.851–856. New York: IEEE.
17. Saab SS, Shen D, Orabi M, et al. Iterative learning control: practical implementation and automation. *IEEE Trans Ind Electron* 2022; 69(2): 1858–1866.
18. Yan Q, Cai J, Ma Y, et al. Robust learning control for robot manipulators with random initial errors and iteration-varying reference trajectories. *IEEE Access* 2019; 7: 32628–32643.
19. Meng T and He W. Iterative learning control of a robotic arm experiment platform with input constraint. *IEEE Trans Ind Electron* 2018; 65(1): 664–672.
20. Chen Y, Chu B and Freeman CT. Generalized iterative learning control using successive projection: algorithm, convergence, and experimental verification. *IEEE Trans Control Syst Technol* 2020; 28(6): 2079–2091.
21. Chi R, Hou Z, Jin S, et al. Computationally efficient data-driven higher order optimal iterative learning control. *IEEE Trans Neural Netw Learn Syst* 2018; 29(12): 5971–5980.
22. Meng T and He W. *Iterative learning control for flexible structures*. Springer, 2020.
23. Rogers E, Chu B, Moore K, et al. Iterative learning control—algorithms, applications and future research directions. In: *Proceedings of the 63rd IEEE conference on decision and control (CDC)*, Milan, Italy, 16–19 December 2024.
24. Barton KL, Hoelzle DJ, Alleyne AG, et al. Cross-coupled iterative learning control of systems with dissimilar dynamics: design and implementation. *Int J Control* 2011; 84(7): 1223–1233.
25. Jiang Z and Chu B. Norm optimal iterative learning control: a data-driven approach. *IFAC Pap Online* 2022; 55(12): 482–487.
26. Drost S, Pustina P, Angelini F, et al. Experimental validation of functional iterative learning control on a one-link flexible arm. In: *2023 IEEE international conference on robotics and automation (ICRA)*, London, UK, 29 May–2 June 2023, pp.5291–5297. New York: IEEE.
27. Zhang N. Design and implementation of walking beam manipulator in automatic production line based on PLC. *J Eur Syst Autom* 2019; 52(2): 173–178.
28. Wang F and Zhu Y. Motion simulation analysis and load analysis of the moving beam system in the walking beam. *Adv Mater Res* 2014; 945–949: 619–625.
29. Nguyen XM, Lawayeb F, Rodríguez Ayerbe P, et al. Nonlinear model predictive control of steel slab walking beam reheating furnace based on a numerical model. In: *Proceedings of the IEEE conference on control applications (CCA)*, Antibes, France, 8–10 October 2014, pp.191–196. New York: IEEE.
30. Jafari H, Castaño M, Gustafsson T, et al. On control structure design for a walking beam furnace. In: *Proceedings of the 25th Mediterranean conference on control and automation (MED)*, Valletta, Malta, 3–6 July 2017, pp.1355–1360. New York: IEEE.
31. Groover MP. *Automation, production systems, and computer-integrated manufacturing*. 5th ed. Pearson India Education Services Pvt. Ltd, 2018.
32. Blanken L, Isil G, Koekebakker S, et al. Data-driven feedforward learning using non-causal rational basis functions: application to an industrial flatbed printer. In: *2018 annual American control conference (ACC)*, Milwaukee, WI, USA, 27–29 June 2018, pp.6672–6677. New York: IEEE.
33. Matijević MS, Lazarević MP, Filipović VŽ, et al. Implementation of the iterative learning control in feedback controlled systems. *Tehnika* 2025; 80(3): 307–314. (in Serbian)
34. Matijević MS, Lazarević MP, Filipović VŽ, et al. Comparison of serial and parallel implementations of ILC in a closed-loop feedback system. In: *12th international conference on electrical, electronics and computer engineering (IcETRAN)*, Čačak, Serbia, 9–12 June 2025.
35. Alfaro V and Vilanova R. Fragility-rings – a graphic tool for PI/PID controllers robustness-fragility analysis. *IFAC Proc Vol* 2013; 45(3): 187–192.
36. Hägglund T. A unified discussion on signal filtering in PID control. *Control Eng Pract* 2013; 21(8): 994–1006.
37. Bošković MČ, Šekara TB, Stojić DM, et al. Novel tuning rules for PIDC controllers in automatic voltage regulation systems under constraints on robustness and sensitivity to measurement noise. *Int J Electr Power Energy Syst* 2024; 157: 109791.
38. Barton KL, van de Wijdeven JJM, Alleyne AG, et al. Norm optimal cross-coupled iterative learning control. In: *Proceedings of the 47th IEEE conference on decision and control (CDC)*, Cancún, Mexico, 9–11 December 2008.