

Research Article

Robust Conditions for Iterative Learning Control in State Feedback and Output Injection Paradigm

Muhammad A. Alsubaie , **Mubarak KH. Alhajri,**
Tarek S. Altowaim, and Salem H. Salamah

Department of Electronics Engineering, College of Technological Studies, PAAET, Kuwait

Correspondence should be addressed to Muhammad A. Alsubaie; ma.alsubaie@paaet.edu.kw

Received 7 October 2018; Accepted 2 January 2019; Published 20 January 2019

Academic Editor: Petko Petkov

Copyright © 2019 Muhammad A. Alsubaie et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A robust Iterative Learning Control (ILC) design that uses state feedback and output injection for linear time-invariant systems is reintroduced. ILC is a control tool that is used to overcome periodic disturbances in repetitive systems acting on the system input. The design basically depends on the small gain theorem, which suggests isolating a modeled disturbance system and finding the overall transfer function around the delay model. This assures disturbance accommodation if stability conditions are achieved. The reported design has a lack in terms of the uncertainty issue. This study considered the robustness issue by investigating and setting conditions to improve the system performance in the ILC design against a system's unmodeled dynamics. The simulation results obtained for two different systems showed an improvement in the stability margin in the case of system perturbation.

1. Introduction

Repetitive systems repeatedly perform a predefined task, $r(t)$, with a fixed time duration, $0 \leq t \leq T$, with high precision [1]. Iterative Learning Control (ILC) is a control technique inspired by human learning methods that use repetition to improve performance. ILC was developed in the mid 1980s [2], where the first introduced control law required system stability within the trial time duration as a condition for error convergence from trial to trial. The impact of ILC on industry can be found in many systems such as those used in pick-and-place operations. One example is a robot arm used to move cans, one at a time, from one moving belt to another without content spillage for an infinite number of trials. Other applications of the ILC theory can be found in chemical patch processes and automated manufacturing plants.

In repetitive systems, ILC uses information provided from previous executions/trials to update the current control input with the purpose of enhancing the performance and accommodating periodic disturbances from trial-to-trial. Thus, a system performs a predefined task, records the input, measures the output, and uses the error signal ($e_k(t) = r - y_k$) as a forcing term to update the new control signal. After each

trial, the system has to reset to its initial position to start the next one. All of the calculations take place in the stoppage time, the time required for the system to reset between trials. One common approach to ILC takes the form $u_{k+1} = u_k + Le_k$, where u is the input, L is the learning gain, e is the error signal, and index k denotes the trial. For more details on ILC theory and applications, refer to [3, 4].

Repetitive control (RC) [5] is another technique used to accommodate periodic disturbances and enhance the performance of repetitive systems without the need for reset between trials; the reference to follow is continuous with $r(t) = r(t + N)$, where N is the number of samples.

It has been found that any periodic signal can be generated by an autonomous system containing a delay model along the forward path with a positive feedback loop [6]. An accommodation of this type of signal can be achieved using the internal model principle by duplicating the system inside a feedback loop. The work introduced in [7, 8] gives the appropriate selection of the required controller, RC or ILC, to accommodate the periodic signal depending on the location of the internal model of the disturbance. The designed framework explicitly incorporates the current error feedback, while in [9] a modified framework was introduced

that incorporated the current error feedback as well as feeding forward the previous error with experimental verification. Basically, the idea in both frames was that a solution for one controller, RC or ILC, was the solution for both.

The novelty of this paper lies with setting new robust conditions for different cases in the ILC design within the framework proposed in [9]. The simulation results obtained showed the reliability of the new design that included unmodeled system dynamics. The results obtained showed the advantage of the new robust conditions over the proposed designs in terms of system perturbation and modeling mismatch.

The following section briefly discusses the ILC design in the general case under the framework proposed in [9]. The design robustness and performance against the unmodeled dynamics of the proposed ILC designs are presented in Section 3. Section 4 considers two examples to illustrate the design advantage, where the first is a 7th-order model of the x -axis of a gantry robot and the second is a Nonminimum Phase Plant (NMP). The simulation results obtained are also considered. The conclusion and future work are discussed in Section 5.

2. State Feedback and Output Injection ILC Background

Let the system to be considered in this paper have m outputs, p inputs, and n states. The discrete linear time-invariant system's overall transfer function in the state space form is given as $P(z) = C(zI_n - A)^{-1}B + D$, where the matrices A, B, C , and D have appropriate dimensions. If the system output is defined as $y(z)$, with $u(z)$ as the input, the output equation is $y(z) = P(z)u(z)$.

Considering a single trial with a finite time duration having N samples, the model of the system dynamics can be expressed as follows:

$$\begin{aligned} x_k(i+1) &= Ax_k(i) + Bu_k(i), \quad x_k(0) = x_0 \\ y_k(i) &= Cx_k(i) + Du_k(i) \end{aligned} \quad (1)$$

where $0 \leq i \leq N-1$. After each trial, the system resets to its initial position. Thus, there is no loss of generality to set $x_0 = 0$.

Many ILC designs rely on expressing the model in trial notation only, rather than using both the time and trial, because the time is fixed for each trial. These designs include the norm optimal ILC [12] and predictive norm optimal ILC [13]. We here introduce the following supervectors:

$$\begin{aligned} u_k &= [u_k(0), u_k(1), \dots, u_k(N-1)]^T \\ y_k &= [y_k(0), y_k(1), \dots, y_k(N-1)]^T \end{aligned} \quad (2)$$

These allow the dynamics for each trial to be written as follows:

$$y_k = Pu_k \quad (3)$$

with

$$P = \begin{bmatrix} D & 0 & 0 & \cdots & 0 \\ CB & D & 0 & \cdots & 0 \\ CAB & CB & D & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N-2}B & CA^{N-3}B & CA^{N-4}B & \cdots & D \end{bmatrix} \quad (4)$$

where the P matrix elements are the Markov parameters. In the same manner, the reference can be defined with the following vector form:

$$r = [r(0), r(1), \dots, r(N-1)]^T \quad (5)$$

Then, the ILC objective is to generate a new input signal for each trial such that the system output follows the reference trajectory with high precision. Many ILC designs can be found in the literature, where one basic choice is to select an input with the form $u_{k+1} = u_k + \Gamma e_k$, where the error vector of trial k is $e_k = r - y_k$ and Γ is the learning gain [4, 14].

A periodic signal with appropriate boundary conditions can be generated by an autonomous system consisting of a positive feedback control loop with a pure time delay in the forward path. Thus, a periodic signal of length N samples in discrete-time can be modeled as follows: (t_k is the time instant of trial k)

$$\begin{aligned} x_w(t_{k+1}) &= A_w x(t_k), \quad x_w(t_0) = x_{w0} \\ w(t_k) &= C_w x(t_k) \end{aligned} \quad (6)$$

with A_w matrix of size $N \times N$ given as

$$A_w = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \quad (7)$$

and the $1 \times N$ row vector C_w as

$$C_w = [1 \ 0 \ 0 \ \cdots \ 0] \quad (8)$$

Then, the control problem can be defined to find a robust controller $K(z)$ (where z denotes the discrete-time delay operator) for the robust periodic control problem as follows.

Given an $m \times l$ transfer-function matrix $P(z)$ with an input vector consisting of both plant and disturbance inputs, $u = u_p + u_w$, the output signal is defined as in (3), and reference signal $r(t_k) = r(t_{k+N})$. It is necessary to design $K(z)$ such that the overall closed loop system is asymptotically stable; the tracking error, $e_k = r - y_k$, tends to zero along the trial domain; and the previous two conditions are robust.

The solution considered in [7, 8] uses the internal model principle [15] and small gain theorem to set stability conditions to design both feedback and observer gains using the Linear Quadratic Regulator (LQR) in the current error

feedback case, where the periodic disturbances act on the system input (ILC). The study [9] considered a more general case that incorporated both the current error and past error in the designed framework.

The work presented in [9] considered two design schemes for ILC. The first used state feedback, and the second used output injection. Each case had two different stability conditions, which depended on using either the current error feedback or past error feedforward. The following subsection briefly explains the design steps in [9] and the stability condition for each case.

2.1. State Feedback-Based ILC. For a single channel, consider the system in (6), which also introduces the following $N \times 1$ vector:

$$B_w = [0 \ \cdots \ 0 \ 0 \ 1]^T \quad (9)$$

and

$$D_w = \begin{cases} 0 & \text{if past error feedforward case} \\ 1 & \text{if current error feedback case} \end{cases} \quad (10)$$

for a more general design. A multi-input multioutput case is considered by defining A_l to be a diagonal matrix consisting of A_w along its diagonal.

$$A_l = \text{diag} \{A_w\} \quad (11)$$

The same is true for B_l, C_l , and D_l , where each diagonal block is repeated p times (acting on the system input). Thus, when considering the periodic problem proposed in Figure 1, the transfer function of $\Phi(z)$ is given as follows:

$$\begin{aligned} & C_l (zI_{N_u} - A_l)^{-1} B_l + D_l \\ &= \begin{cases} (z^N I_{N_u} - I_{N_u})^{-1} & \text{if } D_w = 0 \\ (I_{N_u} - z^{-N} I_{N_u})^{-1} & \text{if } D_w = 1 \end{cases} \quad (12) \end{aligned}$$

The design considered, which uses the state feedback, is found in [9, 16], where the overall idea is to combine both the plant and internal model in one structure in the following form:

$$\begin{aligned} \begin{bmatrix} x_{l,k}(i+1) \\ x_k(i+1) \end{bmatrix} &= \begin{bmatrix} A_l & 0 \\ BC_l & A \end{bmatrix} \begin{bmatrix} x_{l,k}(i) \\ x_k(i) \end{bmatrix} + \begin{bmatrix} B_l \\ BD_l \end{bmatrix} \tilde{u}_k(i), \\ & i = 0, 1, \dots, N-1 \quad (13) \end{aligned}$$

$$\begin{aligned} e_k &= r(i) - [DC_l \ C] \begin{bmatrix} x_{l,k}(i) \\ x_k(i) \end{bmatrix} \\ & - DD_l \tilde{u}_k(i) \end{aligned}$$

Stabilizing this system guarantees the accommodation of periodic disturbances because the output of the combined system is the error, and its input is the control input difference. Note that $x_{l,k}$ is the internal model system state, $\Phi(z)$.

The following can be selected as the control difference input of the combined system: being

$$\tilde{u}(i) = -K_l \begin{bmatrix} x_{l,k}(i) \\ x_k(i) \end{bmatrix} \quad (14)$$

An observer is also introduced to estimate the states for the overall system as in [9]

$$\begin{aligned} & \begin{bmatrix} \hat{x}_{l,K}(i+1) \\ \hat{x}_{K+1}(i+1) \end{bmatrix} \\ &= \begin{bmatrix} A_l & 0 \\ BC_l & A \end{bmatrix} \begin{bmatrix} \hat{x}_{l,k}(i) \\ \hat{x}_k(i) \end{bmatrix} - \begin{bmatrix} B_l \\ BD_l \end{bmatrix} K \begin{bmatrix} \hat{x}_{l,k}(i) \\ \hat{x}_k(i) \end{bmatrix} \\ & + L([DC_l \ C] - DD_l K_l) \left(\begin{bmatrix} \hat{x}_{l,k}(i) \\ \hat{x}_k(i) \end{bmatrix} - \begin{bmatrix} x_{l,k}(i) \\ x_k(i) \end{bmatrix} \right) \end{aligned} \quad (15)$$

The difference between the system estimated above and the actual system will lead to a more simplified structure for describing the overall feedback path dynamics:

$$\begin{aligned} \begin{bmatrix} \tilde{x}_{l,K}(i+1) \\ \tilde{x}_{K+1}(i+1) \end{bmatrix} &= \begin{bmatrix} A_l & 0 \\ BC_l & A \end{bmatrix} \begin{bmatrix} \tilde{x}_{l,k}(i) \\ \tilde{x}_k(i) \end{bmatrix} \\ & - \begin{bmatrix} B_l \\ BD_l \end{bmatrix} K_l \begin{bmatrix} \tilde{x}_{l,k}(i) \\ \tilde{x}_k(i) \end{bmatrix} \end{aligned} \quad (16)$$

where ($\tilde{*} = \hat{*} - *$). The feedback gain K_l can now be designed via the LQR by solving the well-known Riccati equation. Refer to [9] for more details on the design.

Stability conditions can be found by isolating the internal model $z^N I_p$ of the internal model and expressing the overall system in terms of its input and output as $H(z)$. Then, a sufficient condition for stability can be written as follows:

$$\|H(z)\| < 1 \quad (17)$$

where the $H(z)$ equation differs depending on the error case considered, past error feedforward, or current error feedback. The following can be written for **past error feedforward case**:

$$H(z) = (G(z) + P(z))G(z)^{-1} \quad (18)$$

and for **current error feedback**

$$H(z) = G(z)(G(z) + P(z))^{-1} \quad (19)$$

where $G(z)$ in both cases is governed by the following:

$$\begin{aligned} G(z) &= [DC_l \ C] \left(zI_{N_{p+n}} - \begin{bmatrix} A_l & 0 \\ BC_l & A \end{bmatrix} + \begin{bmatrix} B_l \\ DB_l \end{bmatrix} K_l \right)^{-1} \\ & \times \begin{bmatrix} B_l \\ DB_l \end{bmatrix} + DD_l \end{aligned} \quad (20)$$

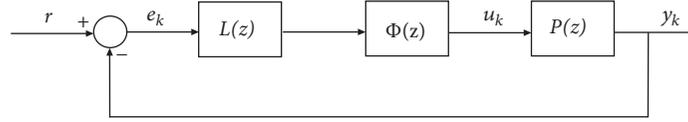


FIGURE 1: ILC as a feedback problem [9].

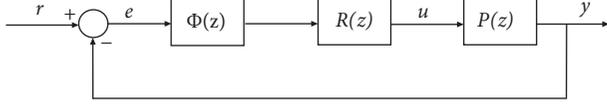


FIGURE 2: RC as a feedback problem [9].

2.2. *Output Injection-Based ILC.* We start from the dual repetitive structure given in [9] with a single internal model and build up a combined system between the plant and internal model for a repetitive feedback problem, as shown in Figure 2. we define an estimator for the overall system with a periodic disturbance assumed to effect the plant at its input. The structure is manipulated to cancel the disturbance signal via the use of the estimator output as a correction term. This will lead to an ILC dual system in the following form:

$$\begin{bmatrix} x_{l,k}(i+1) \\ x_k(i+1) \end{bmatrix} = \begin{bmatrix} A_l & 0 \\ BC_l & A \end{bmatrix} \begin{bmatrix} x_{l,k}(i) \\ x_k(i) \end{bmatrix} + \begin{bmatrix} L_l \\ BD_l \end{bmatrix} \tilde{u}_k(i) \quad (21)$$

Again, this system is derived to form the overall structure, (22), which is used to design the estimator gain for both error cases, past error feedforward, and current error feedback, as follows: $\tilde{*} = \hat{*} - *$.

$$\begin{bmatrix} \tilde{x}_{l,k}(i+1) \\ \tilde{x}_k(i+1) \end{bmatrix} = \begin{bmatrix} A_l & 0 \\ BC_l & A \end{bmatrix} \begin{bmatrix} \tilde{x}_{l,k}(i) \\ \tilde{x}_k(i) \end{bmatrix} - \begin{bmatrix} L_l \\ \hat{L}_l \end{bmatrix} \times [DC_l \ C] \begin{bmatrix} \tilde{x}_{l,k}(i) \\ \tilde{x}_k(i) \end{bmatrix} \quad (22)$$

The estimator gain design relies on forming the overall structure to fit the LQR design to accommodate the periodic disturbances acting on the system input. This structure was shown in [9] to compromise both current and past error incorporations with a stability condition that suggested isolating the internal model, $z^{-N}I_p$, and finding the overall system that governs its output to its input, as given earlier in (17). Here, $H(z)$ differs depending on the error case to include. Thus, the following can be written for the **past error feedforward case**:

$$H(z) = G(z) (G(z) - \hat{L}(z))^{-1} \quad (23)$$

and for **current error feedback**

$$H(z) = (G(z) - I_m) (G(z) - I_m - \hat{L}(z))^{-1} \quad (24)$$

where $G(z)$ in both cases is governed by the following

$$\begin{aligned} G(z) = & [DC_l \ C] \left(zI_{N_p+n} - \begin{bmatrix} A_l & 0 \\ BC_l & A \end{bmatrix} \right. \\ & \left. + \begin{bmatrix} L_l \\ \hat{L}_l \end{bmatrix} [DC_l \ C] \right)^{-1} \times \begin{bmatrix} C_l \\ 0 \end{bmatrix} + D_l \end{aligned} \quad (25)$$

and $\hat{L}(z) = C_l B_l z^{-1} + C_l A_l B_l z^{-2} + \dots + C_l A_l^{N-1} B_l z^{-N}$

The next section investigates the robustness issue and sets the appropriate condition to ensure system stability in the presence of a modeling mismatch for each of the cases reported earlier for the ILC design.

3. Robust ILC Design with Both State Feedback and Output Injection

This section investigates the robustness property of the two designed ILC controllers found in [9] with past error feedforward and current error feedback using the stability condition assigned in (17). The previous reported works did not discuss this issue, which forms the main novelty of the work presented in this paper.

We start with the stability condition in (17) and consider the following cases:

- (i) Past error feedforward in state feedback design.

The starting point is the stability condition given in (17) where the induced norm has to be less than 1 to guarantee system stability. A more conservative restriction is to consider the singular values instead; thus the condition will be as follows:

$$\bar{\sigma}(H(z)) < 1 \quad (26)$$

which clearly indicates that all the eigenvalues of $H(z)$ are inside the unit circle once the maximum singular value is considered. Verifying this condition in the maximum case ensures reference tracking and periodic disturbance accommodation. Now, consider the case where unmodeled system dynamics or system uncertainty defined as (Δ) act on the system in operation. To investigate this, define $[P = P_o + P_o \Delta W]$, where P_o, Δ, W are the nominal plant, uncertainty, and uncertainty weight, respectively, where each of the defined variables is stable, causal, and linear time invariant for simplicity. In combination with the

definition of $H(z)$ given in (18), we can write the following derivation:

$$\bar{\sigma}(G + P_o + P_o\Delta W) < \bar{\sigma}(G) \quad (27)$$

$$\underline{\sigma}(G + P_o) + \bar{\sigma}(P_o\Delta W) < \bar{\sigma}(G) \quad (28)$$

The selection in the previous equation is the only selection that guarantees the maximum possible singular values without violating the condition in (27). Taking the uncertainty part on one side and the other parts to the right side yields the following:

$$\bar{\sigma}(P_o\Delta W) < \bar{\sigma}(G) - \underline{\sigma}(G) - \underline{\sigma}(P_o) \quad (29)$$

Maintaining that 20 is true and maximizing the left hand side will give the possible variation in the system dynamics on the right hand side and sets the upper bound for the system to avoid unwanted performance through the operation. This can be found if the right hand side is of the form $\bar{\sigma}(G) - \underline{\sigma}(G) - \bar{\sigma}(P)$ as pointed out earlier. To extend the previous property and set a weight for the uncertainty that gives a better upper bound and allows the system to deal with unmodeled dynamics throughout its operation, (29) can be manipulated to have the following form:

$$\bar{\sigma}(\Delta) < \frac{\bar{\sigma}(G) - \underline{\sigma}(G) - \underline{\sigma}(P_o)}{\bar{\sigma}(W)\bar{\sigma}(P_o)} \quad (30)$$

Maximizing the left hand side of (30), such that the right hand side is kept to the minimum, can be seen as solving the following:

$$\bar{\sigma}(\Delta) < \frac{\min}{\max} = \frac{\bar{\sigma}(G) - \underline{\sigma}(G) - \underline{\sigma}(P_o)}{\sigma(W)\bar{\sigma}(P_o)} \quad (31)$$

Now, $\bar{\sigma}(G) \neq \underline{\sigma}(G)$ unless G is a scalar multiplied by the identity, which is not the case. Further investigation of weight (W) is taken into account and can be expressed, based on the fact that $\bar{\sigma}(\Delta) < 1$, in the following:

$$1 < \frac{\bar{\sigma}(G) - \underline{\sigma}(G) - \underline{\sigma}(P_o)}{\sigma(W)\bar{\sigma}(P_o)} \quad (32)$$

$$\bar{\sigma}(W) < \frac{\bar{\sigma}(G) - \underline{\sigma}(G) - \underline{\sigma}(P_o)}{\bar{\sigma}(P_o)} < 1$$

The above condition will set the upper limit for the weighting factor such that the uncertainty property is extended.

(ii) Current error feedback in state feedback design.

Start again with the stability condition given in (17), where the induced norm has to be less than 1. We again consider the system uncertainty to be (Δ) acting on the system in operation. Define [$P = P_o +$

$P_o\Delta W$] as above with the same properties considered in the past-error case. Following the same steps, in combination with the definition of $H(z)$ given in (19), we can write the following derivation in terms of the singular values:

$$\bar{\sigma}\left(\frac{G}{G - P_o - P_o\Delta W}\right) < 1 \quad (33)$$

which leads to writing the above after manipulation as follows:

$$\bar{\sigma}(\Delta) > \frac{\bar{\sigma}(G) - \underline{\sigma}(G) - \underline{\sigma}(P_o)}{\sigma(W)\bar{\sigma}(P_o)} \quad (34)$$

Again, assuming that (34) is true, and because the uncertainty is assumed to be stable ($\bar{\sigma}(\Delta) < 1$), (34) can be written as follows:

$$\frac{\bar{\sigma}(G) - \underline{\sigma}(G) - \underline{\sigma}(P_o)}{\bar{\sigma}(W)\bar{\sigma}(P_o)} < 1 \quad (35)$$

Equation (35) will give the appropriate condition for weighting factor (W) such that the left hand side is minimized:

$$1 > \bar{\sigma}(W) > \frac{\bar{\sigma}(G) - \underline{\sigma}(G) - \underline{\sigma}(P_o)}{\bar{\sigma}(P_o)} \quad (36)$$

Condition (36) is the same as that in (32) to a limit degree. Equation (36) sets the lower limit to the weight selection, whereas (32) sets the upper limit to the uncertainty weight, which has a wider and better range than that of (36). This result supports the experimental results obtained for the past error feedforward case in ILC instead of current error feedback in [9], where the past error feedforward case has a more reliable design against system perturbation.

(iii) Past error feedforward in output injection design.

As in the two previous robustness analyses, this analysis starts from the stability condition, which should have a norm of less than 1, and the definition given for ($H(z)$) in (23). Using the singular value properties, we can start by writing the following:

$$\bar{\sigma}\left(\frac{G}{G - \hat{L}}\right) < 1 \quad (37)$$

considering the uncertainty (Δ) to be added to the nominal plant such as [$G = G_o + G_o\Delta W$], where G_o is the nominal representation of the combined system that is defined in (25). Thus, we can write (37) as follows:

$$\bar{\sigma}\left(\frac{G_o + G_o\Delta W}{G_o + G_o\Delta W - \hat{L}}\right) < 1 \quad (38)$$

$$\sigma(G_o + G_o\Delta W) < \bar{\sigma}(G_o + G_o\Delta W - \hat{L})$$

Now, assuming that (38) is true, this can be derived to give the necessary condition for the uncertainty limit as follows:

$$\begin{aligned}\sigma(G_o + G_o\Delta W) &< \bar{\sigma}(G_o + G_o\Delta W - \dot{L}) \\ \sigma(G_o + G_o\Delta W) &\leq \sigma(G_o) + \sigma(G_o)\bar{\sigma}(\Delta)\bar{\sigma}(W) \\ &< \sigma(G_o) + \sigma(G_o)\bar{\sigma}(\Delta)\bar{\sigma}(W) \\ &\quad - \bar{\sigma}(\dot{L}) \\ \sigma(G_o) &< \left[\sigma(G_o) + \frac{\sigma(\dot{L})}{1 + \sigma(\Delta)\sigma(W)} \right]\end{aligned}\quad (39)$$

Equation (39) can be formed as follows:

$$\sigma(G_o) - \sigma(G_o) < \frac{\sigma(\dot{L})}{1 + \sigma(\Delta)\sigma(W)} \quad (40)$$

Since G is a combined system consisting of the internal model and plant, with $\bar{\sigma}(G_o) \neq \underline{\sigma}(G_o)$, then, the uncertainty effect can be maximized such that the upper bound is minimized as in (41)

$$\bar{\sigma}(\Delta) < \frac{\bar{\sigma}(\dot{L}) - \bar{\sigma}(G_o) + \underline{\sigma}(G_o)}{[\bar{\sigma}(G_o) - \underline{\sigma}(G_o)]} \quad (41)$$

This will lead to setting the upper boundary for weighting factor (W) as follows:

$$\bar{\sigma}(W) < \frac{\underline{\sigma}(G_o)}{1 - \underline{\sigma}(G_o)} \quad (42)$$

Equation (42) is always positive and less than 1. This can extend the robustness of the system against the unmodeled dynamics. The fourth case is expected to have the condition in (42) as a result of extending the system uncertainty range.

(iv) Current error feedback in output injection design.

Following the same starting point and steps to derive the required condition, the resulting equation to set the limit of the weighting factor such that the uncertainty will not drive the system to the instability stage is given by the following:

$$\sigma(\Delta) > \frac{\sigma(\dot{L}) - \bar{\sigma}(G_o) + \underline{\sigma}(G_o)}{[\bar{\sigma}(G_o) - \underline{\sigma}(G_o)]} \quad (43)$$

and

$$1 > \bar{\sigma}(W) > \frac{\underline{\sigma}(G_o)}{1 - \underline{\sigma}(G_o)} \quad (44)$$



FIGURE 3: Gantry robot benchmark [10].

This clearly gives the upper limit of the weighting factor, which does not exceed the maximum of 1 and is not less than the limit set in (44).

The next section presents simulation results obtained on two systems that show the performance improvement against the system uncertainty and modeling mismatch when considering the robust design over the previously reported design framework for the ILC case.

4. ILC Design Simulation Results

In this section, simulation results are obtained for two different examples, where the first is the x -axis of a gantry robot and the second is a nonminimum phase plant, to verify the design success in suppressing a dynamic matrix change in systems with difficult mathematical structures.

Example 1. The gantry robot is a multi-axis test facility that was developed to implement different ILC theories and verify their success in a real experimental environment (see Figure 3). The benchmark has three axes that are orthogonal to each other, and fixed above one end of a 6m long chain conveyor to simulate a real industrial production line. This represents a real control problem because of the nature of the operation, where the gantry has to perform a pick-and-place operation in synchronization with a moving belt without ripples or sudden movements during the operation. This task is very difficult to control because the synchronization between the conveyor speed and payload placement position has to be made with high precision. Check [17] for more details on the gantry. A sampling frequency of 100Hz was used to define the gantry axes dynamical response models. In this paper, only the 7th-order x -axis representation is used to verify the robust theory development. The gantry itself has been the benchmark for several reported ILC implementations such as those in [10, 18, 19].

The x -axis was modeled using the data obtained from a frequency response test, which resulted in the following 7th-

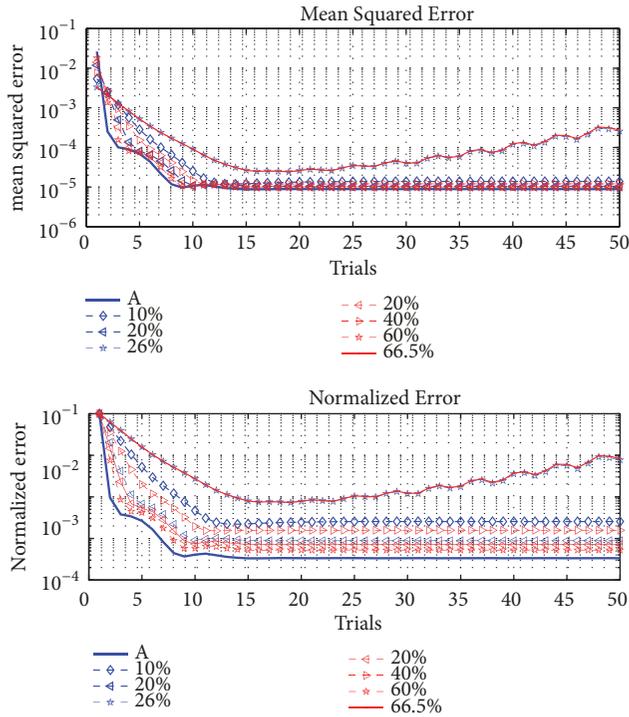


FIGURE 4: x -axis mean squared error for the two different cases, with and without $w.f.$

order continuous-time transfer-function:

$$P(s) = \frac{13077183.4436(s + 113.4)}{s(s^2 + 61.57s + 1.125 \times 10^4)} \times \frac{(s^2 + 30.28s + 2.13 \times 10^4)}{(s^2 + 227.9s + 5.647 \times 10^4)(s^2 + 466.1s + 6.142 \times 10^5)} \quad (45)$$

For a complete system description, refer to [20].

The x -axis is sampled at 100Hz and a reference of $4s$ is considered, which generates 400 sample points to deal with in each trial. The design steps considered are those of the output injection case with past error feedforward, where two different cases are considered. In the first, no weighting factor is used. The second uses the weighting factor limit found in (32), which is 0.7820. Thus, one choice is to select a weighting factor that is half of the limit, which is the case in this example. The system had been operated for 50 trials in a simulation, and the obtained results are as follows.

Figure 4 shows the mean squared error for the two cases, where the blue line represents the case where no weighting factor is considered, with 10%, 20%, and a 26% changes in the system dynamical matrix applied. The red line is the case where a weighting factor of $w.f. = 0.3910$ is applied, which was less than the limit found in (32). Dynamical changes of 20%, 40%, 60%, and 66.5% are considered, and the figure clearly shows the advantage of using the weighting factor to avoid unmodeled dynamics to a better level. Figure 5 shows the case with the maximum allowed change in system dynamics and how the error will start building up after it converges to better levels. The use of the weighting factor extends the

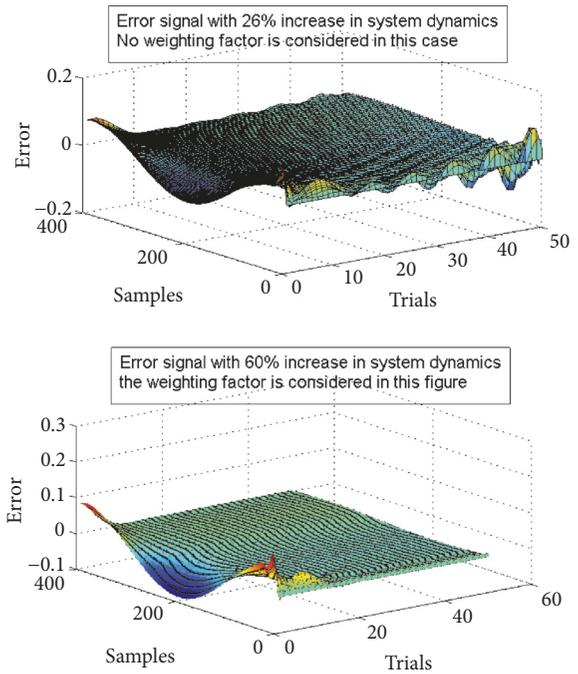


FIGURE 5: The output signal for the x -axis in two different cases, with and without $w.f.$



FIGURE 6: The nonminimum phase plant experimental test facility [11].

maximum allowed change in mismodeling compared to the case where no weighting factor is applied.

Example 2. In this example, a NMP plant was tested against dynamical matrix changes under the absence/presence of the weighting factor. The physical system was constructed to implement both ILC and repetitive controller (RC) schemes, which made it possible to verify reported results such as those in [11, 21]. This paper reports simulation results that support the idea of extending the level of system mismodeling using the weighting factor. The NMP plant has one zero in the right half plane, which makes this system hard to test using the ILC scheme because of the instability associated with plant inversion. As a result, any sudden change in system dynamics would result in an instable response. The mathematical equation that describes the system shown in Figure 6 is given as follows:

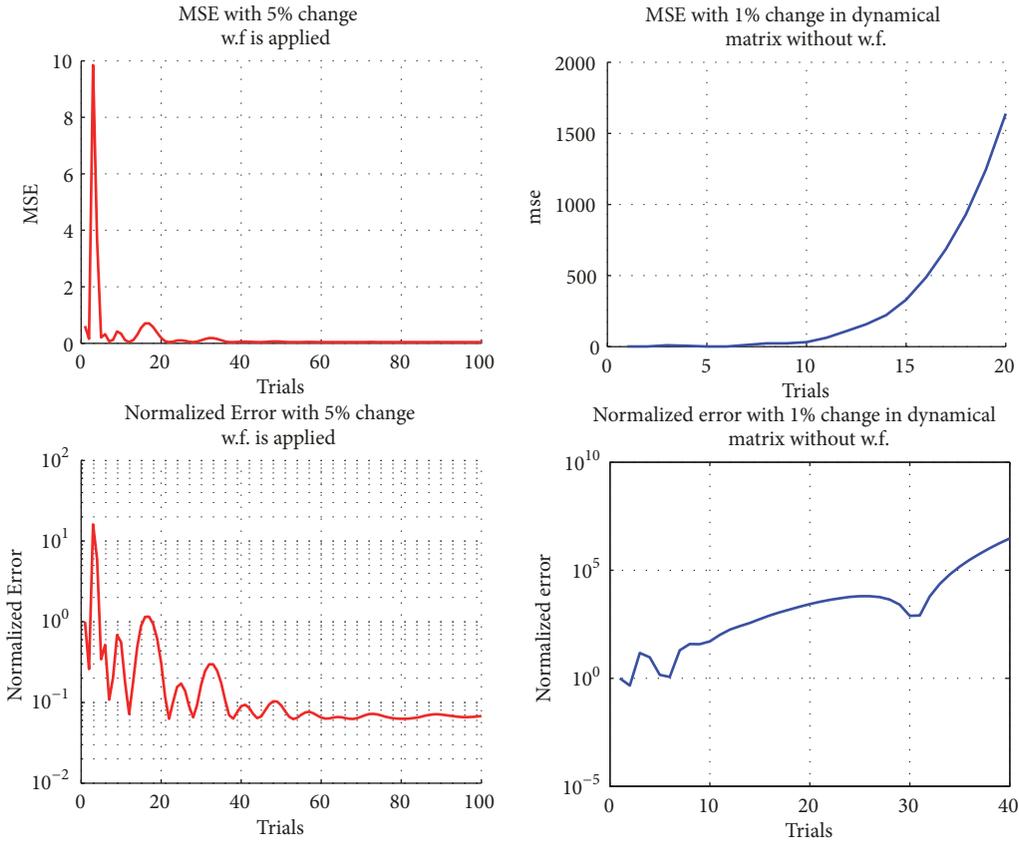


FIGURE 7: Mean squared error and the normalized error for the NMP output with and without the presence of w.f.

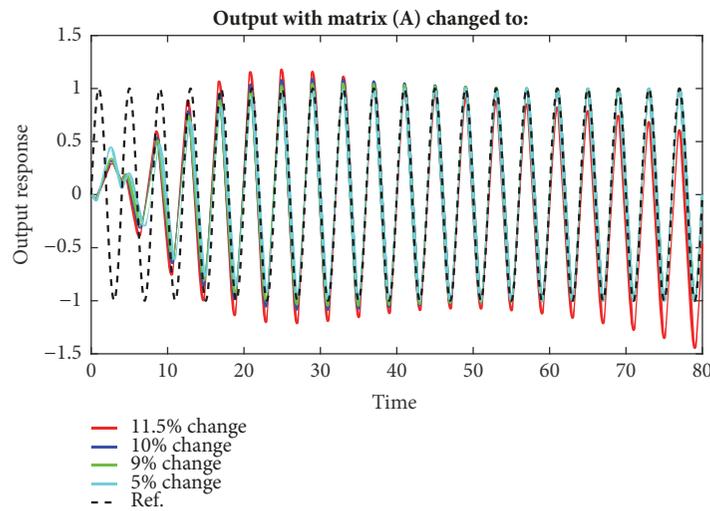


FIGURE 8: Output responses with dynamic matrix (A) variations and *wf* applied.

$$P(s) = \frac{1.202(4-s)}{s(s+9)(s^2+12s+56.25)} \quad (46)$$

The facility consists of a rotary mechanical system of inertias, dampers, torsional springs, a timing belt, pulleys, and gears. A further spring-mass-damper system is connected to the input in order to increase the relative degree

and complexity of the system. For further details on the test facility, refer to [11]

This system was tested using two different cases: one that ignored the weighting factor and one that considered it. A sampling frequency of 100Hz was selected, and a reference of 4s was applied, generating 400 sample points to record in each trial. Figure 7 shows the two different

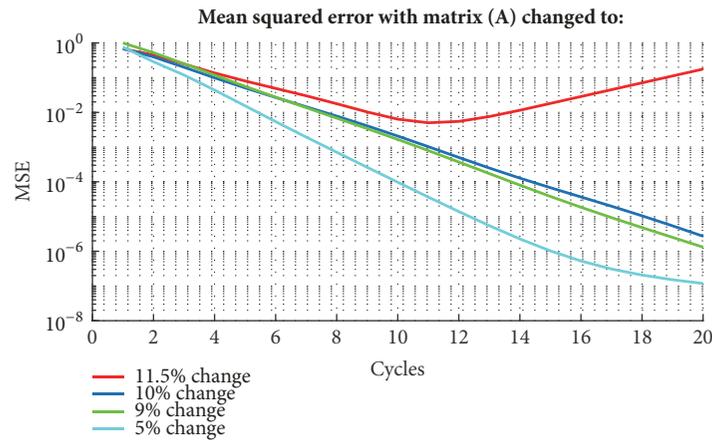


FIGURE 9: Mean squared error for the NMP output with dynamic matrix (A) variations and wf applied.

cases, where the red color shows the mean squared error and normalized error for a change of up to 5% in the system dynamical matrix with the presence of the weighting factor. In the second case, the blue color shows a fast error divergence with a 1% change in the system dynamical matrix without the presence of the weighting factor. As the results show, the weighting factor had a direct impact on the design procedure by providing a better stability region with better level of reference tracking against model mismatching compared to the case where the weighting factor was ignored.

Figure 8 shows different output responses with dynamic matrix (A) variations, where the weighting factor is used, and it can be seen that the output has an extended dynamic matrix change margin compared to the case where no weighting factor applied. Figure 9 shows the mean square error against dynamic matrix (A) variations with the use of the weighting factor. The figure ensures the advantage of using the weighting factor to enhance model mismatch case.

5. Conclusions and Future Work

In this paper, conditions were set to extend the range of linear system uncertainty based on the singular value principle. Different cases were discussed, and conditions were found that extended the system robustness against the system unmodeled dynamics. The simulation results obtained verified the success of the weighting factor in extending the range of uncertainty considered with the ILC design via state feedback and output injection. A high level of reference tracking was achieved for a system uncertainty of up to 66.5% for the 7th-ordered model of a gantry robot and up to 5% for the NMP model. Experimental results are expected to verify the developed conditions to overcome system perturbation in the future. Future work will examine the effect of the new obtained conditions to extend the range of uncertainty on the system performance for a dual repetitive controller. The conditions to suppress the effect of noise on the system performance will also be investigated.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] E. Rogers, K. Galkowski, and D. H. Owens, *Control systems theory and applications for linear repetitive processes*, vol. 349 of *Lecture Notes in Control and Information Sciences*, Springer, Berlin, Germany, 2007.
- [2] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.
- [3] H.-S. Ahn, K. L. Moore, and Y. Chen, *Iterative learning control: robustness and monotonic convergence for interval systems*, Springer Science & Business Media, 2007.
- [4] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control: a learning-based method for high-performance tracking control," *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, 2006.
- [5] S. Hara, T. Omata, and M. Nakano, "Synthesis of repetitive control systems and its application," in *Proceedings of the 1985 24th IEEE Conference on Decision and Control*, vol. 24, pp. 1387–1392, Fort Lauderdale, FL, USA, December 1985.
- [6] T. Inoue, M. Nakano, T. Kubo, S. Matsumoto, and H. Baba, "High accuracy control of a proton synchrotron magnet power supply," in *Proceedings of the 8th IFAC World Congress*, Kyoto, Japan, 1981.
- [7] D. de Roover and O. H. Bosgra, "Dualization of the internal model principle in compensator and observer theory with application to repetitive and learning control," in *Proceedings of the 1997 American Control Conference*, vol. 6, pp. 3902–3906, Albuquerque, NM, USA, June 1997.
- [8] D. de Roover, O. H. Bosgra, and M. Steinbuch, "Internal-model-based design of repetitive and iterative learning controllers for

- linear multivariable systems,” *International Journal of Control*, vol. 73, no. 10, pp. 914–929, 2000.
- [9] C. T. Freeman, M. A. Alsubaie, Z. Cai, E. Rogers, and P. L. Lewin, “A common setting for the design of iterative learning and repetitive controllers with experimental verification,” *International Journal of Adaptive Control and Signal Processing*, vol. 27, no. 3, pp. 230–249, 2013.
- [10] L. Hladowski, K. Galkowski, Z. Cai, E. Rogers, C. T. Freeman, and P. L. Lewin, “Experimentally supported 2D systems based iterative learning control law design for error convergence and performance,” *Control Engineering Practice*, vol. 18, no. 4, pp. 339–348, 2010.
- [11] C. T. Freeman, P. L. Lewin, and E. Rogers, “Experimental evaluation of iterative learning control algorithms for non-minimum phase plants,” *International Journal of Control*, vol. 78, no. 11, pp. 826–846, 2005.
- [12] N. Amann, D. H. Owens, and E. Rogers, “Robustness of norm-optimal iterative learning control,” in *Proceedings of the UKACC International Conference on Control (Conf. Publ. No. 427)*, pp. 1119–1124, Exeter, UK, 1996.
- [13] N. Amann, D. H. Owens, and E. Rogers, “Predictive optimal iterative learning control,” *International Journal of Control*, vol. 69, no. 2, pp. 203–226, 1998.
- [14] H. Ahn, Y. Q. Chen, and K. L. Moore, “Iterative learning control: Brief survey and categorization,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1099–1121, 2007.
- [15] B. A. Francis and W. M. Wonham, “The internal model principle for linear multivariable regulators,” *Applied Mathematics and Optimization. An International Journal with Applications to Stochastics*, vol. 2, no. 2, pp. 170–194, 1975.
- [16] C. Freeman, P. Lewin, E. Rogers, D. Owens, and J. Hatonen, “An Optimality-Based Repetitive Control Algorithm for Discrete-Time Systems,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 1, pp. 412–423, 2008.
- [17] J. D. Ratcliffe, *Iterative learning control implemented on a multi-axis system [Ph.D. thesis]*, School of Electronics and Computer Science, University of Southampton, 2005.
- [18] J. D. Ratcliffe, T. J. Harte, J. J. Hatonen, P. L. Lewin, E. Rogers, and D. H. Owens, “Practical implementation of a model inverse optimal iterative learning controller on a gantry robot,” in *Proceedings of the IFAC Workshop on Adaptation and Learning in Control and Signal Processing (ALCOSP 04) and IFAC Workshop on Periodic Control Systems (PSYCO 04)*, pp. 687–692, Yokohama, Japan, 2004.
- [19] C. T. Freeman, Z. Cai, E. Rogers, and P. L. Lewin, “Objective-driven ilc for point-to-point movement tasks,” in *Proceedings of the 2009 American Control Conference*, pp. 252–257, St. Louis, MO, USA, June 2009.
- [20] J. Ratcliffe, L. van Duinkerken, P. Lewin et al., “Fast norm-optimal iterative learning control for industrial applications,” in *Proceedings of the 2005, American Control Conference, 2005.*, pp. 1951–1956, Portland, OR, USA, 2005.
- [21] Z. Cai, C. T. Freeman, P. L. Lewin, and E. Rogers, “Iterative learning control for a non-minimum phase plant based on a reference shift algorithm,” *Control Engineering Practice*, vol. 16, no. 6, pp. 633–643, 2008.



Hindawi

Submit your manuscripts at
www.hindawi.com

