Repetitive Process Based Higher-order Iterative Learning Control Law Design

Robert Maniarski*[†], Wojciech Paszke, Bing Chu[‡], Eric Rogers

07.04.2023

Abstract

This paper uses the repetitive process setting to develop new results on the design of higher-order ILC control laws. The basic idea of higher-order ILC is to use information from a finite number of previous trials, as opposed to just the last trial, to update the control input to be applied on the next trial, with the primary objective of improving the error convergence performance. The sufficient conditions ensuring the convergence of the resulting control scheme are established with repetitive process setting and through utilizing non-unit memory repetitive process models. Also, the corresponding control law gains are derived from a set of linear matrix inequality constraints. Finally, a practical example is used to demonstrate the properties of the new design.

1 Introduction

The reference trajectory is repetitive in many applications, such as robotics or chemical batch systems. The repetition of the trajectory allows the application of a feedforward-type control known as iterative learning control (ILC). In turn, this allows the control input for each repetition to be a function of previous trial data to improve performance from trial to trial. In ILC, it is the input, a signal, that is updated rather than the controller, which is a system.

Each repetition is known as a trial, and the sequence of operations is that a trial is completed, the system resets to the starting position, and then the subsequent trial begins, either immediately after the resetting is complete or after a further time has elapsed since completion of the resetting. Different from repetitive control, ILC can be applied to systems that operate over a finite duration, and then there is a stoppage time before the next one of the same duration begins, and so on. The duration of a trial is known as the trial length.

*This research was funded in whole or in part by National Science Centre in Poland, grant No. 2020/39/B/ST7/01487.

[†]R. Maniarski and W. Paszke are with the Institute of Automation, Electronic and Electrical Engineering, University of Zielona Góra, ul. Szafrana 2, 65-246 Zielona Góra, Poland. E-mail: {r.maniarski, w.paszke}@iee.uz.zgora.pl

[‡]B. Chu and E. Rogers are with the School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK. E-mail: {b.chu,etar}@ecs.soton.ac.uk

The objective of ILC design is to construct the input such that the corresponding output precisely tracks a reference signal specified over a finite time interval. Given a reference vector, the error on each trial can be formed as the difference between this vector and the trial output vector, also known as the trial profile vector. The ILC design problem is to force this sequence of trial errors to converge with an increasing trial number and ensure acceptable performance along the trials. A novel feature is the availability to the control law of temporal information from the previous trial that would be non-causal outside the ILC setting, e.g., for discrete dynamics at sample instant p information at $p + \lambda$, $\lambda > 0$, can be used provided it has been generated on a previous trial.

Since the original work, widely credited to [2], ILC has remained a significant area of control systems research with many algorithms experimentally verified in the research laboratory and applied in industrial applications. An overview of the early developments can be found in, e.g., the survey papers [1, 3]. The survey paper [11] focuses on run-to-run control in the chemical process industries. This area is one where there is a stoppage time between one trial and the next instead of resetting. Applications areas include industrial robotics, for early application see, e.g., [7], nano-positioning, for recent progress see, e.g., [**?**] and optimizing broiler weight in agriculture [**?**]. Also, there have been productive work on using ILC in healthcare. For examples of recent progress in this last area, see, e.g., [**?**, **?**].

As the trial length is finite, the values of a discrete variable along a trial can be assembled into a column vector, termed a super-vector, in the literature. The outcome is that a standard difference equation governs the trial-to-trial updating of the error for linear. Hence, all associated analyses from standard linear systems theory can be applied. Given the finite trial length trial to trial error convergence is possible even if the system is not stable (all system state matrix eigenvalues inside the open unit circle in the complex plane). Therefore, it is necessary to design a stabilizing feedback control law in such cases and apply ILC to the resulting dynamics.

An alternative setting for analysis is provided by formulating the dynamics as a 2D system, where the directions of information propagation are from trial to trial and along the trial, respectively. Repetitive processes are a distinct class of 2D systems where information propagation in one direction only occurs over a finite duration, see [10] and hence a close match with ILC dynamics. The stability theory for these processes has led to ILC laws that have been designed and experimentally validated for the first results on this latter aspect, see [5]. This setting allows the single-step design of an ILC law for error convergence and regulation of the dynamics along the trials.

In ILC on a particular trial, information from all previous trials is available for use in design. In particular, higher-order ILC uses information from a finite number $M > 1$ of previous trials in forming the control input for the subsequent trial. The use of such information is the subject area of this paper. By utilizing repetitive process setting and some matrix transformation techniques, the sufficient conditions for the existence the ILC law are derived by a set of LMI constraints, ensuring that the resulting ILC scheme ensures the tracking error convergence. Also, the proposed design is extended to cases where the system dynamics is strictly proper. Finally, the feasibility and effectiveness of the proposed method are demonstrated by an example.

The following notation is used in this paper; the identity and null matrices of compatible dimensions are denoted, respectively, by I and 0. Also the notation $M \succ 0$ (respectively $M \prec 0$) denotes a symmetric positive definite (respectively negative definite) matrix. The symbol sym $\{M\}$ is a shorthand notation for $M + M^T$. Moreover, $\rho(\cdot)$ denotes the spectral radius of its matrix arguments and $\otimes$ stands for the matrix Kronecker product. The symbol diag$\{M_1, M_2, \cdots, M_n\}$ denotes a block diagonal matrix with diagonal blocks $M_1, M_2, \cdots, M_n$. Finally, the new results in this paper are formulated in terms of LMIs and hence the following lemmas are useful in transforming non-LMI formulations into LMI form.

Lemma 1 *[4] Given matrices $\Gamma = \Gamma^T \in \mathbb{R}^{p\times p}$ and two matrices Λ, Σ of column dimension p, there exists a matrix W that satisfies*

$$\Gamma + \mathrm{sym}\{\Lambda^T W \Sigma\} \prec 0, \tag{1}$$

if, and only if

$$\Lambda^{\perp^T} \Gamma \Lambda^{\perp} \prec 0, \text{ and } \Sigma^{\perp^T} \Gamma \Sigma^{\perp} \prec 0, \tag{2}$$

where $\Lambda^{\perp}$ and $\Sigma^{\perp}$ are arbitrary matrices whose columns form a basis of nullspaces of Λ and Σ respectively. This means that $\Lambda\Lambda^{\perp} = 0$ and $\Sigma\Sigma^{\perp} = 0$.

2 Higher-order ILC for Linear Discrete-time Systems

Consider the discrete time-invariance linear state-space system model

$$\begin{aligned} x_k(p+1) =& A x_k(p) + B u_k(p), \\ y_k(p) =& C x_k(p), \end{aligned} \tag{3}$$

where the subscript $k \geq 0$ denotes the trial number and p denotes the discrete-time variable such that $0 \leq p \leq \alpha - 1$, with $\alpha < \infty$. This last parameter denotes the number of samples along a trial (α times the sampling period denotes the trial length). Furthermore, $x_k(t) \in \mathbb{R}^n$ is the state vector, $y_k(t) \in \mathbb{R}^m$ is the output vector, and $u_k(t) \in \mathbb{R}^m$ is the control input vector, and it is assumed that the system resets to the same initial state vector x_o on each trial, where no loss of generally occurs in assuming $x_o = 0$.

Additionally, it is assumed that the system in (3) has relative degree $\kappa > 1$, i.e., the first $\kappa - 2$ Markov parameters are the null matrix and that for $\kappa - 1$ is nonsingular. Specifically, for all $i < \kappa - 1$ $CA^iB = 0$ and $CA^{\kappa-1}B \neq 0$.

Also, let us define the output tracking error on the current trial by

$$e_k(p) = y_{ref}(p) - y_k(p), \forall k \geq 1,\ p \in [0, \alpha - 1], \tag{4}$$

where $y_{ref}(p)$ denotes the desired trajectory that is invariant with respect to trials (iterations). Now, let us suppose that a standard form of ILC law (i.e. the means of updating the control vector from trial-to-trial) given as

$$u_{k+1}(p) = u_k(p) + \Delta u_k(p) \tag{5}$$

is applied to a system represented by (3) where $\Delta u_k(p)$ is the control input update. This form of control law constructs the input for the next trial ($k+1$) as the sum of the previous trial input (k) and $\Delta u_k(p)$, where this last term is computed using the previous trials error. Specifically, $\Delta u_k(p)$ is defined as a function of the error and/or control vectors on a finite number (denoted here as M) of previous trials since these data are collected and stored in memory. Therefore, the control law correction term $\Delta u_k(p)$ in (5) is defined as

$$\Delta u_k(p) = K\eta_{k+1}(p+1) + \sum_{j=1}^{M} K_{j-1} e_{k+1-j}(p+\kappa), \tag{6}$$

where K and K_{j-1}, $j = 1, 2, \ldots, M$ are matrices of compatible dimensions to be designed. Also, $\eta_{k+1}(p+1)$ is a vector-valued variable such that

$$\eta_{k+1}(p+1) = x_{k+1}(p) - x_k(p).$$

It is important to note that the form of (6) is used to compensate for the influence of κ. As seen the second term in the ILC law, i.e., (6), has an anticipatory gain operator form as described in [8]. In particular, $\Delta u_{k+1}(p)$ at sample instant p is paired with $e_{k+1-j}(p+\kappa)$ at sample instant $p+\kappa$ for all $j = 1, 2, \ldots, M$. Moreover, the terms $e_{k+1-j}(p+\kappa)$ are available for computations since the error vector from the already completed trials $k+1-j$ are available.

Remark 1 *Parts of the analysis that follows are based on the z-transform. See, e.g., [3] for details of why this transform is applicable given that the trial length is finite.*

Application of the z-transform gives

$$e_{k+1-j}(p+\kappa) = z^{\kappa} e_{k+1-j}(p+\kappa)$$

and hence, it can be shown that

$$z^{\kappa}(zI - \mathcal{A})^{-1} = \sum_{i=0}^{\kappa-1} z^{\kappa-1-i}\mathcal{A}^i + \mathcal{A}^{\kappa}(zI - \mathcal{A})^{-1}.$$

Next, apply the control law (6) and apply the same steps as in [8] to write the controlled dynamics as

$$\begin{aligned} \eta_{k+1}(p+1) =& \mathcal{A}\eta_{k+1}(p) + \sum_{j=1}^{M} \mathcal{B}_{j-1} e_{k+1-j}(p), \\ e_{k+1}(p) =& \mathcal{C}\eta_{k+1}(p) + \sum_{j=1}^{M} \mathcal{D}_{j-1} e_{k+1-j}(p), \end{aligned} \tag{7}$$

where

$$\begin{aligned} &\mathcal{A}{=}A{+}BK,\ \mathcal{B}_0{=}BK_0,\ \mathcal{B}_j{=}BK_j,\ \mathcal{D}_j{=}CBK_j, \\ &\mathcal{C} = -CA^{\kappa-1}(A{+}BK),\ \mathcal{D}_0{=}I{-}CA^{\kappa-1}BK_0 \end{aligned} \tag{8}$$

and $j = 1, \ldots, M-1$. This state-space model has the form of a non-unit memory linear repetitive process. Such processes make a series of sweeps, termed trials in this paper, through a set of dynamics defined over a finite duration. Once a trial is complete, the process resets to the starting location, and the next trial can begin. The 2D systems structure arises from the fact that the output on any trial acts as a forcing function and contributes to the next trial's dynamics. The control problem that standard systems designs cannot remove is that the sequence of outputs can contain oscillations that increase in amplitude with the number of trials.

In (7) it is the more general case where the previous $M > 1$ previous trial outputs that contribute to the current trial output. The background to repetitive processes, including their application to modeling physical processes, can be found in [10] and the relevant references cited in this publication. Moreover, a stability theory has been developed and is used in this paper for

ILC design, where in (7) $\eta_{k+1}(p)$ is the current trial state vector and $e_{k+1}(p)$ is the current trial output. The simplest form boundary conditions are

$$x_{k+1}(0) = d_{k+1},\ k \geq 1, y_{1-j}(p) = Y_{1-j}(p),$$

where $j = 1, 2, \ldots, M$, d_{k+1} is an $n \times 1$ vector with known constant entries and the vectors $Y_{1-j}(p)$ have entries that are known functions of p over the trial length. No loss of generality arises from setting $d_{k+1} = 0,\ k \geq 1$.

In [10], a stability theory for linear repetitive processes is developed, which given the unique control problem, requires that a bounded initial trial profile produces a sequence of bounded trial outputs, where boundedness is defined in terms of the norm on the underlying function space. This stability theory distinguishes between the property over the finite and fixed trial length, termed asymptotic stability, and uniformly, i.e., for all possible values of the (finite) trial length, termed stability along the trial. Moreover, this theory can be applied to non-unit memory examples, as detailed next.

Introduce the matrix

$$\bar{\mathcal{D}} = \begin{bmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \ddots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & I \\ \mathcal{D}_{M-1} & \mathcal{D}_{M-2} & \mathcal{D}_{M-3} & \cdots & \mathcal{D}_0 \end{bmatrix}. \tag{9}$$

The next result is a version of the results presented in [10] extended to a non-unit memory case and gives a the corresponding stability conditions.

Lemma 2 *The discrete non-unit memory linear repetitive process described by* (7) *and* (8) *is stable along the trial if and only if,*

i) $\rho(\bar{\mathcal{D}}) < 1$,

ii) $\rho(\mathcal{A}) < 1$,

iii) all eigenvalues of the transfer-function matrix $\bar{G}(z)$ *have modulus strictly less than unity* $\forall |z| = 1$

where

$$\bar{G}(z) = \begin{bmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \ddots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & I \\ G_M(z) & G_{M-1}(z) & G_{M-2}(z) & \cdots & G_1(z) \end{bmatrix}$$

and

$$G_j(z) = \mathcal{C}(zI - \mathcal{A})^{-1}\mathcal{B}_{j-1} + \mathcal{D}_{j-1},\ 1 \leq j \leq M.$$

This last result is difficult to apply directly to the design of the ILC law matrices. In particular, checking the condition *iii*) of the above lemma requires computations at all points on the boundary of the unit circle in the complex plane. This fact, in turn, introduces difficulties in stability checking and transforming into design procedures for control law matrices in (6). A possible way to overcome these problems is as follows, starting from rewriting the model of (7) in unit memory form as

$$\begin{aligned} \eta_{k+1}(p+1) =& \mathcal{A}\eta_{k+1}(p) + \bar{\mathcal{B}}\bar{e}_k(p), \\ \bar{e}_{k+1}(p) =& \bar{\mathcal{C}}\eta_{k+1}(p) + \bar{\mathcal{D}}\bar{e}_k(p), \end{aligned} \tag{10}$$

where $\bar{\mathcal{D}}$ is defined in (9) and

$$\begin{aligned} &\bar{e}_k(p) = \begin{bmatrix} e_{k-M+1}^T(p) & \cdots & e_{k-1}^T(p) \; e_k^T(p) \end{bmatrix}^T, \\ &\bar{\mathcal{B}} = \begin{bmatrix} \mathcal{B}_{M-1} & \cdots & \mathcal{B}_1 \cdots \mathcal{B}_0 \end{bmatrix}, \; \bar{\mathcal{C}} = \begin{bmatrix} 0 & 0 & \cdots & 0 & \mathcal{C}^T \end{bmatrix}^T. \end{aligned}$$

Remark 2 *In the unit memory case, for ease of presentation only, this last model has the structure of a controlled discrete linear repetitive process with no current trial input where $\eta_{k+1}(p)$ is the current trial state vector and $e_{k+1}(p)$ is the current trial output vector. Condition i) of the previous result will guarantee convergence of the error sequence (the core ILC design requirement) over the finite trial length but this condition is independent of the state dynamics, which govern the performance along the trials. Condition ii) of this last result regulates the dynamics along the current trial, and condition iii) enforces error convergence for all possible finite trial lengths. (In the case of $\kappa = 1$ condition i) of this last result requires that $\rho(I - CBK_0) < 1$, and hence not possible if $CB = 0$.)*

The state-space model (10) gives the z-transform model of the dynamics

$$\bar{e}_{k+1}(z) = G(z)\bar{e}_k(z)$$

where $G(z) = \bar{\mathcal{C}}(zI - \mathcal{A})^{-1}\bar{\mathcal{B}} + \bar{\mathcal{D}}$. Moreover, the entries in the transfer-function matrix $G(z)$ that govern the trial-to-trial error convergence performance are those in the last block row since

$$e_{k+1}(z) = [G_{M-1}(z), G_{M-2}(z), \cdots, G_0(z)]\bar{e}_k(z),$$

where $G_{j-1}(z) = \mathcal{C}(zI - \mathcal{A})^{-1}\mathcal{B}_{j-1} + \mathcal{D}_{j-1}, \; 1 \le j \le M$.

Given the unit memory repetitive process description of (10), the problem of selecting required control law matrices K and K_{j-1}, $j = 1, 2, \ldots, M$ in (6) can be formulated as an LMI-based stability condition (at the expense of sufficient but not necessary conditions for stability along the trial). Introduce the matrix $\Phi = \text{diag}\{1, -1\}$ and then the following lemma gives an LMI-based sufficient condition for stability along the trial of processes described by (10), where this result is from [**?**].

Lemma 3 *A unit memory linear repetitive process described by* (10) *is stable along the trial if there exist compatibly dimensioned $P_1 \succ 0$, $P_2 \succ 0$ such that*

$$\begin{bmatrix} \mathcal{A} & I \\ \bar{\mathcal{C}} & 0 \end{bmatrix} (\Phi \otimes P_1) \begin{bmatrix} \mathcal{A} & I \\ \bar{\mathcal{C}} & 0 \end{bmatrix}^T + \begin{bmatrix} \bar{\mathcal{B}} & 0 \\ \bar{\mathcal{D}} & I \end{bmatrix} (\Phi \otimes P_2) \begin{bmatrix} \bar{\mathcal{B}} & 0 \\ \bar{\mathcal{D}} & I \end{bmatrix}^T \prec 0 \tag{11}$$

is feasible.

Clearly, the result of Lemma 3 cannot be directly applied for the considered ILC design since there exist product terms between control law matrices and matrix variables P_1 and P_2. Additionally, given Remark 1, it is required to impose structural constraints on P_2 since the bottom block row of $G(z)$ is of critical interest. Therefore, the matrix variable P_2 has the structure

$$P_2 = \operatorname{diag}\{P_{21}, \gamma^2 I\}, \tag{12}$$

where $P_{21} \succ 0$ and γ is a positive scalar satisfying $0 < \gamma \leq 1$. Given this structure of P_2 the following constraint can be imposed

$$\|G_{M-1}(z)\, G_{M-2}(z)\, \cdots\, G_0(z)\|_\infty < \gamma.$$

3 LMI-based Controller Design

This section uses the stability results for unit memory linear repetitive processes [10] to develop a new ILC design algorithm. Specifically, introducing additional slack matrix variables decouples the product terms between control law matrices and matrix variables P_1 and P_2 and additionally provides additional design options. The condition stated in Lemma 3 is further modified to provide results for characterizing the stability along the trial that is also well suited for the control law design of the corresponding ILC law.

Theorem 1 *Let γ be a positive scalar satisfying $0 < \gamma \leq 1$. Also, suppose that an ILC law* (6) *is applied to a discrete linear system represented by* (3). *Then the resulting ILC scheme described as a unit memory discrete linear repetitive process of the form* (10) *is stable along the trial, and hence trial-to-trial error convergence occurs, if there exist compatibly dimensioned matrices $P_1 \succ 0$, $P_{21} \succ 0$, W_1, W_2, F_1, F_2, F_3 and a scalar β such that $|\beta| < 1$ and the following matrix inequality is feasible*

$$\begin{bmatrix} \Upsilon_1 - \operatorname{sym}\{W_1\} & (\star) & (\star) \\ \Upsilon_3 + \Psi W_1^T - \beta W_1 & \Upsilon_2 + \operatorname{sym}\left\{\beta W_1 \Psi^T\right\} & (\star) \\ F_b - W_2 & -F_a^T + W_2 \Psi^T & P_2 - \operatorname{sym}\{F_3\} \end{bmatrix} \prec 0, \tag{13}$$

where P_2 is of the form in (12) *and*

$$\Psi = \begin{bmatrix} \mathcal{A} & \bar{\mathcal{B}} \\ \bar{\mathcal{C}} & \bar{\mathcal{D}} \end{bmatrix}, \Upsilon_1 = \begin{bmatrix} P_1 & 0 \\ 0 & 0 \end{bmatrix}, \Upsilon_2 = \begin{bmatrix} -P_1 & 0 \\ 0 & -P_2 \end{bmatrix},$$
$$\Upsilon_3 = \begin{bmatrix} 0 & F_1 \\ 0 & F_2 \end{bmatrix}, F_a = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}, F_b = \begin{bmatrix} 0 & F_3 \end{bmatrix}. \tag{14}$$

Proof 1 *Assume that the inequality defined in* (13) *has a feasible solution. Then, it can be rewrit-*

ten as (1) *with*

$$\begin{aligned}\Gamma &= \begin{bmatrix} \Upsilon_1 & \Upsilon_3^T & F_b^T \\ \Upsilon_3 & \Upsilon_2 & -F_a \\ F_b & -F_a^T & P_2 - sym\{F_3\} \end{bmatrix}, \Lambda^T = \begin{bmatrix} I & 0 \\ \beta I & 0 \\ 0 & I \end{bmatrix}, \\ \mathcal{W} &= \begin{bmatrix} W_1 \\ W_2 \end{bmatrix}, \Sigma = \begin{bmatrix} -I & \Psi^T & 0 \end{bmatrix}. \end{aligned} \tag{15}$$

Next, by Lemma 1, the inequality (13) *is solvable for* W *if and only if the inequality* (2) *holds. Select*

$$\Sigma_\perp = \begin{bmatrix} \Psi^T & 0 \\ I & 0 \\ 0 & I \end{bmatrix}, \ \Lambda_\perp = \begin{bmatrix} \beta I \\ -I \\ 0 \end{bmatrix}.$$

and the first inequality in (2) *becomes*

$$\begin{bmatrix} (\beta^2-1)P_1 & 0 \\ 0 & -P_2 \end{bmatrix} + sym\left\{ \begin{bmatrix} 0 \\ -\beta I \end{bmatrix} \begin{bmatrix} F_1 & F_2 \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \right\} \prec 0. \tag{16}$$

Hence

$$\Gamma \leftarrow \begin{bmatrix} (\beta^2-1) & 0 \\ 0 & -P_2 \end{bmatrix}, \Lambda^T \leftarrow \begin{bmatrix} 0 \\ -\beta I \end{bmatrix}, W \leftarrow \begin{bmatrix} F_1 & F_2 \end{bmatrix}, \Sigma \leftarrow \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

and by Lemma 1 with $\Lambda_\perp = [I \ 0]^T$ *and noting that* ${\Sigma^{\perp}}^T \Gamma \Sigma^{\perp}$ *vanishes,* (16) *reduces to* $(\beta^2 - 1)P_1 \prec 0$. *This last inequality holds for any* $|\beta| < 1$ *and* $P_1 \succ 0$. *Next, with the notation of* (15) *the second inequality in* (2) *can be rewritten as*

$$\begin{aligned} &\begin{bmatrix} \mathcal{A}P_1\mathcal{A}^T - P_1 & \mathcal{A}P_1\bar{\mathcal{C}}^T & 0 \\ \bar{\mathcal{C}}P_1\mathcal{A}^T & \bar{\mathcal{C}}P_1\bar{\mathcal{C}}^T - P_2 & 0 \\ 0 & 0 & P_2 \end{bmatrix} \\ &+ sym\left\{ \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} \begin{bmatrix} \bar{\mathcal{B}}^T & \bar{\mathcal{D}}^T & -I \end{bmatrix} \right\} \prec 0. \end{aligned} \tag{17}$$

Next, assign

$$\begin{aligned} &\Gamma \leftarrow \begin{bmatrix} \mathcal{A}P_1\mathcal{A}^T - P_1 & \mathcal{A}P_1\bar{\mathcal{C}}^T & 0 \\ \bar{\mathcal{C}}P_1\mathcal{A}^T & \bar{\mathcal{C}}P_1\bar{\mathcal{C}}^T - P_2 & 0 \\ 0 & 0 & P_2 \end{bmatrix}, W \leftarrow \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix}, \\ &\Lambda^T \leftarrow I, \ \Sigma \leftarrow \begin{bmatrix} \bar{\mathcal{B}}^T & \bar{\mathcal{D}}^T & -I \end{bmatrix} \end{aligned}$$

and by Lemma 1, feasibility of (17) *implies that the second inequality in* (2) *can only hold since, as detailed above, the first inequality in* (2) *vanishes. Finally, the last inequality is equivalent to* (13), *and by Lemma 3, stability along the trial is ensured, and the proof is complete.*

The result of Theorem 1 is not an LMI since it involves bilinear terms arising from a product of the matrices W_1, W_2 and the control law matrices K and K_{j-1}, $j = 1, 2, \ldots, M$ in (6). These difficulties are removed by the result below.

Theorem 2 *Let γ be a positive scalar satisfying $0 < \gamma \leq 1$. Also, suppose that an ILC law* (6) *is applied to a discrete linear system represented by* (3). *Then the resulting ILC scheme described as a unit memory discrete linear repetitive process of the form* (10) *is stable along the trial, and hence trial-to-trial error convergence occurs, if there exist compatibly dimensioned matrices $P_1 \succ 0$, $P_{21} \succ 0$, W_1, N, F_1, F_2, F_3 and a scalar β such that $|\beta| < 1$ and the following LMI is feasible*

$$\begin{bmatrix} \Upsilon_1 - \mathrm{sym}\{W_1\} & (\star) & (\star) \\ \Upsilon_3 + \mathbb{A}W_1^T + \mathbb{B}N - \beta W_1 & \Upsilon_2 + \mathrm{sym}\left\{\beta(\mathbb{A}W_1^T + \mathbb{B}N)^T\right\} & (\star) \\ F_b - [0\ I]W_1 & -F_a^T + [0\ I](\mathbb{A}W_1^T + \mathbb{B}N)^T & P_2 - \mathrm{sym}\{F_3\} \end{bmatrix} \prec 0, \tag{18}$$

where Υ_1, Υ_2, Υ_3, F_a and F_b are as in (14), *P_2 is of the form in* (12), *and*

$$\mathbb{A} = \left[\begin{array}{c|c} A & 0\,0\,0\ldots0 \\ \hline 0 & 0\,I\,0\ldots0 \\ \vdots & 0\,0\,I\,\ddots\,0 \\ \vdots & 0\,0\,0\,\ddots\,0 \\ 0 & 0\,0\,0\ldots I \\ CA^{\kappa-1} & 0\,0\,0\ldots I \end{array}\right], \quad \mathbb{B} = \left[\begin{array}{c} B \\ \hline 0 \\ 0 \\ \vdots \\ 0 \end{array}\right]. \tag{19}$$

If the LMI in this last result is feasible, the following formula gives the corresponding controller matrices of (6)

$$K = [K\ K_0\ K_1\ \ldots\ K_{M-1}] = NW_1^{-1}. \tag{20}$$

Proof 2 *Assume that the LMI* (18) *holds. Also, it follows immediately that the feasibility of* (18) *implies that W_1 is non-singular and hence it is possible to compute K in* (20). *The remaining part follows as for Theorem 1 where Ψ is rewritten as*

$$\Psi = \left[\begin{array}{c|c} \mathcal{A} & \mathcal{B} \\ \hline \mathcal{C} & \mathcal{D} \end{array}\right] = \mathbb{A} + \mathbb{B}K.$$

Then make the change of variables as $N = KW_1$. Additionally, introduce the constraint on the multiplier W_2 as $W_2 = [0\ I]W_1$. This choice introduces conservativeness but still has some additional freedom by introducing the slack matrix variables F_1, F_2, and F_3. The proof is complete.

4 Case study

This section presents a numerical example demonstrating the new design's applicability. The example used is of one axis of a gantry robot used in previous research to verify ILC designs. Further information on this robot is given in [5, 9], where the following transfer function for one axis has been constructed from measure frequency response data

$$G(s) = \frac{15.8869(s + 850.3)}{s(s^2 + 707.6s + 3.377 \times 10^5)}.$$

Sampling using the zero-order-hold method at a frequency of $100[Hz]$, the resulting z-transfer function incorporates a one-sample delay. This delay is generated by the zero-order hold function, which is included in the real-time control card. Consequently, the following transfer function is used for design purposes

$$G(z) = \frac{0.00036482(z^2 + 0.09791z + 0.005951)}{z(z-1)(z^2 + 0.005922z + 0.0008451)}$$

and hence in (3)

$$A = \begin{bmatrix} 0.9941 & 0.0406 & 0.027 & 0 \\ 0.125 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 \\ 0 & 0 & 0.25 & 0 \end{bmatrix}, B = \begin{bmatrix} 0.0625 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$
$$C = \begin{bmatrix} 0 & 0.0467 & 0.0183 & 0.0044 \end{bmatrix}.$$

In this last model $\kappa = 2$ (relative degree two). Let $\beta = 0.01$ and consider the cases when $M = 1$, $M = 2$ and $M = 3$. Then applying Theorem 2 to minimize γ gives the following data

- $M = 1$

$$K = [-17.4730 \ -1.4033 \ -0.4320 \ -0.0000],$$
$$K_0 = 2754.6029,$$
$$\|G_0(z)\|_\infty = 0.0050,$$

- $M = 2$

$$K = [-17.4730 \ -1.4033 \ -0.4320 \ -0.0000],$$
$$K_0 = 2740.9036, \ K_1 = -12.5174,$$
$$\|G_1(z)\ G_0(z)\|_\infty = 0.0046,$$

- $M = 3$

$$K = [-17.4730 \ -1.4033 \ -0.4320 \ -0.0000],$$
$$K_0 = 2740.9040, \ K_1 = -0.0071, \ K_2 = -8.5506,$$
$$\|G_2(z)\ G_1(z)\ G_0(z)\|_\infty = 0.0031$$

The reference trajectory for the system considered is which has been selected for a pick and place robotic task with trial length 2 secs. From Fig 2, it follows that the higher order ILC increases flexibility during design, and it is possible to optimize some variables. Hence higher-order ILC law outperforms the $M = 1$ case. To compare this new design with previously reported alternatives, the design described in [8] results in the following control law matrices

$$K = [-16.3275 \ -0.8546 \ -0.4320 \ 0.0000], \ K_0 = 733.9686$$

and in [6] gives

$$K = [-17.0918 \ -1.2237 \ -0.4331 \ 0.0000], \ K_0 = 641.7524$$

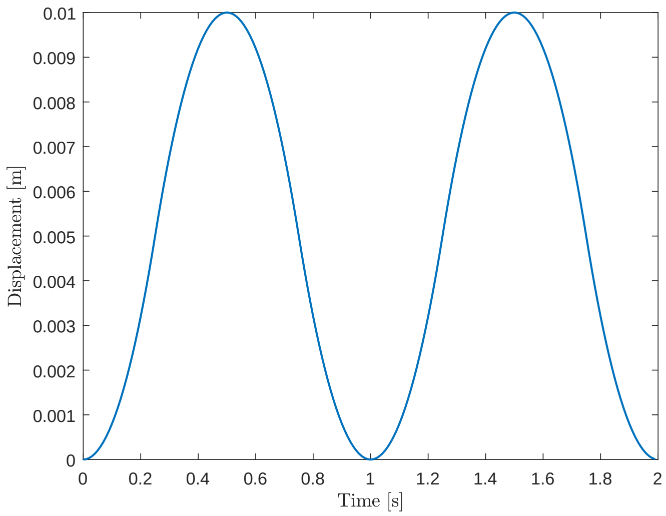

Figure 1: The reference trajectory for the considered axis.

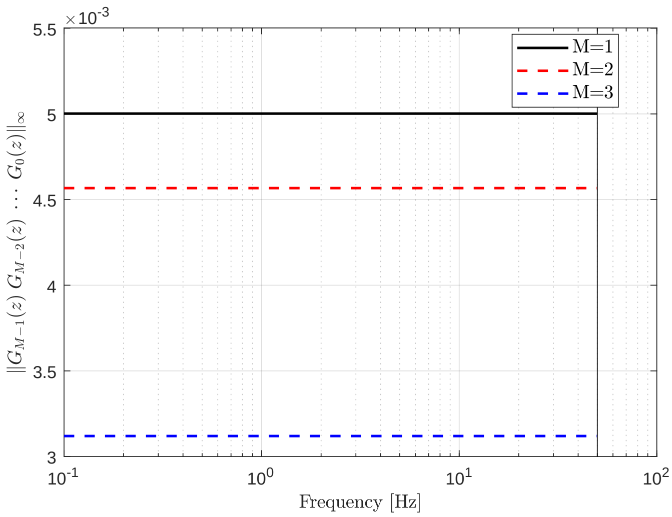

Figure 2: $\mathcal{H}_\infty$ norm of the bottom row in $G(z)$.

The controlled dynamics for each of these designs, the controlled dynamics were simulated over 20 trials. For each, the Euclidean norm of the tracking error was calculated. From Fig. 3 it can be deduced that the new ILC design can lead to more effective controllers which provide faster monotonic error convergence than known alternatives. However, when convergence speed is considered, using a higher-order ILC law does not improve this performance aspect. Further investigation could, as suggested previously in the literature, could lead to improvements in robustness to noise and modeling uncertainties.

5 Conclusions and Further Research

This paper is devoted to the design of higher-order ILC laws for linear discrete systems of repetitive nature. The repetitive process setting has been utilized to formulate the controller design conditions in terms of LMIs and hence they can be easily handled by existing numerical software. A numerical

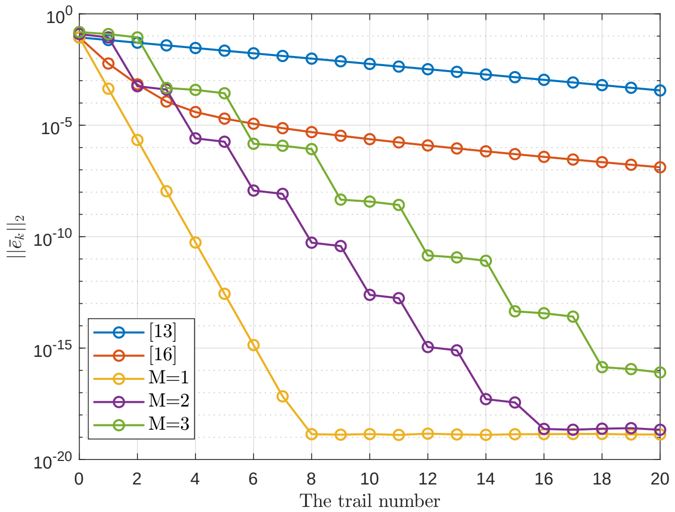

Figure 3: Tracking error over 20 trials.

example illustrates the effectiveness of the design. Future research efforts would be in applying this approach to system models with uncertainty and non-repetitive disturbance. It is also valuable to investigate the inclusion of $\mathcal{H}_\infty/\mathcal{H}_2$ performance measures against non-repetitive disturbances. Finally, the primary aim is to perform a comprehensive experimental verification.

References

[1] H.-S. Ahn, Y.-Q. Chen, and K. L. Moore. Iterative learning control: brief survey and categorization. *IEEE Transactions on Systems, Man and Cybernetics, Part C*, 37(6):1109–1121, 2007.

[2] S. Arimoto, S. Kawamura, and F. Miyazaki. Bettering operation of robots by learning. *Journal of Robotic Systems*, 1(2):123–140, 1984.

[3] D. A. Bristow, M. Tharayil, and A.G. Alleyne. A survey of iterative learning control. *IEEE Control Systems Magazine*, 26(3):96–114, 2006.

[4] P. Gahinet and P. Apkarian. A linear matrix inequality approach to $\mathcal{H}_\infty$ control. *International Journal of Robust and Nonlinear Control*, 4:421–448, 1994.

[5] L. Hładowski, K. Gałkowski, 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*, 18(4):339–348, 2010.

[6] R. Maniarski, W. Paszke, H. Tao, and E. Rogers. Design of ILC laws with conditions for stabilizing linear 2D discrete Roesser models. In *61st IEEE Conference on Decision and Control (CDC)*, pages 1491–1496, 2022.

[7] M. Norrlöf. An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Transactions on Robotics and Automation*, 18(2):245–251, 2002.

[8] W. Paszke, E. Rogers, and K. Gałkowski. Design of iterative learning control schemes for systems with zero Markov parameters. In *54th IEEE Conference on Decision and Control (CDC)*, pages 6083–6088, 2015.

[9] J. D. Ratcliffe, J. J. Hatonen, P. L. Lewin, E. Rogers, T. J. Harte, and D. H. Owens. P-type iterative learning control for systems that contain resonance. *International Journal of Adaptive Control and Signal Processing*, 19:769796, 2005.

[10] E. Rogers, K. Gałkowski, and D. H. Owens. *Control Systems Theory and Applications for Linear Repetitive Processes*, volume 349 of *Lecture Notes in Control and Information Sciences.* Springer-Verlag, Berlin, Germany, 2007.

[11] Y. Wang, F. Gao, and F.J. Doyle. Survey on iterative learning control, repetitive control, and run-to-run control. *Journal of Process Control*, 19(10):1589–1600, 2009.