CYBERNETICS AND PHYSICS, VOL. 15, NO. 1, 2026, 23–30

ACCELERATED ITERATIVE LEARNING CONTROL OF MULTI-AGENT DISCRETE SYSTEMS

Anton Koposov, Pavel Pakshin
Arzamas Polytechnic Institute of
R.E. Alekseev Nizhny Novgorod State Technical University,
ul. Kalinina 19, Arzamas, 607227 Russia
koposov96@yandex.ru, PakshinPV@gmail.com

Article history:
Received 08.04.2026, Accepted 19.06.2026

Abstract

Iterative learning control (ILC) applies to systems that operate in repetitive mode. The goal of control is to ensure that the system's output follows a reference trajectory with a specified accuracy. In ILC literature, each repetition is called a trial, and its finite duration is known as the trial length (also, the terms "pass" and "iteration" are commonly used). With each repetition, ILC should reduce the difference between the reference trajectory and the output signal called trial-to -trial error , ideally bringing it to zero. In this paper, a new ILC design method is proposed for multi-agent discrete systems based on the application of gradient optimization within 2D models in combination with the method of vector Lyapunov functions for repetitive processes. The new method yields a simple ILC structure that accelerates the convergence of the trial-to-trial error. An example confirming this property is provided.

Key words

Iterative learning control (ILC), repetitive processes, multi-agent systems, convergence acceleration, gradient optimization, vector Lyapunov functions, linear matrix inequalities (LMI).

1 Introduction

Iterative learning control (ILC) applies to systems that repeatedly execute the same finite-duration task. In ILC literature, each repetition is called a trial, and its finite duration is known as the trial length (also, the terms "pass" and "iteration" are commonly used). Once each trial is complete, all information generated (state, output, and input) is available to update the control input for the subsequent trial. In ILC, the control input is an updated signal, unlike the controller or system in other approaches.

The early research in ILC are starting from pioneering work [Arimoto et al., 1984], focused on robotic applications. Consider, for example, a gantry robot executing a pick-and-place operation: the robot collects a payload from a specified location, transfers it over a finite duration (the trial length), place it at a fixed location, and then returns to the starting location to repeat this sequence. The survey papers [Bristow et al., 2006; Ahn et al., 2007] overview developments following this initial research. Recent advances can be found, e.g., in [Rogers et al., 2023]. The idea of ILC was known before than [Arimoto et al., 1984] was published, but for various reasons it did not gain popularity. See [Bristow et al., 2006] for more details. The application areas covered ILC include high-precision multilayer laser deposition [Lim et al., 2017; Sammons et al., 2019] and many others. In addition, note productive research on applications in healthcare, including robotic-based rehabilitation of stroke patients [Freeman et al., 2012], and ventricular-support devices [Ketelhut et al., 2019]. Many of these engineering applications have supporting experimental results and clinical testing results for stroke rehabilitation [Meadmore et al., 2014].

Recently, there has been increased interest in robotic networks control algorithms [Konovalov and Matveev, 2025; Sharma and Lather, 2024] and references therein. In this area, the theory of multi-agent systems has become widespread. The ILC approach has also been developed for multi-agent systems; see [Liu and Jia, 2012; Meng et al., 2015; Pakshin et al., 2018; Hock and Schoellig, 2019; Yu et al., 2022; Koposov and Pakshin, 2023] and references therein.

Increasing the trial-to-trial convergence rate is a crucial part of ILC design. In recent works, classical optimization methods [Polyak, 1987] have been successfully applied to ILC design problems: Nesterov's method in [Gu et al., 2019], and the heavy ball method in [Pakshin et al., 2024; Zhang et al., 2024].

In this paper, a new ILC design method is obtained for multi-agent discrete systems based on the application of gradient optimization within 2D models in combination with the method of vector Lyapunov functions, previously proposed by the authors for repetitive processes. This method yields a simple ILC structure that accelerates the convergence of the trial-to-trial error. An example confirming this property is provided.

2 Problem Statement

Consider a network system consisting of N identical linear subsystems (agents) that repeatedly execute an operation, defined equally for all agents. The dynamics of agent i on trial k are described by the discrete state-space model

$$\begin{aligned} x_i(k,p+1) &= Ax_i(k,p) + Bu_i(k,p),\\ y_i(k,p) &= Cx_i(k,p),\\ &i \in [1,N],\ k \geq 0,\ p \in [0,T-1], \end{aligned} \tag{1}$$

where k is the trial number; $x_i(k,p) \in \mathbb{R}^{n_x}$ is the state vector; $u_i(k,p) \in \mathbb{R}$ is the scalar control input, $y_i(k,p) \in \mathbb{R}$ is the scalar output (trial profile); p is the sample number on trial k; and $T < \infty$ is the number of samples on each trial (the same for all i and k). The boundary condition $x_i(k,0) = x_0$ is known, the same for all i and k. By assumption, all entries of the state vector of each agent are measurable. It is also assumed that $CB \neq 0$. This assumption holds in the widespread case when system (1) is equivalent discrete model of a continuous system.

Agents shall track a desired (reference) output trajectory $y_{ref}(p), 0 \leq p \leq T-1, y_{ref}(0) = Cx_0$. Only some network agents (leaders) have direct access to $y_{ref}(p)$. The other agents (followers) shall track the outputs of their associated agents.

The connections between agents are represented by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, 2, \ldots, N\}$ and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ are the sets of its vertices and edges, respectively. The access of agent i to the output of agent j is determined by the edge directed from vertex j to vertex i and is indicated by the ordered pair $(j,i) \in \mathcal{E}$. The elements of the adjacency matrix $\mathcal{S}(\mathcal{G}) = [s_{ij}]_{i,j=1}^{N}$ are defined as follows: $s_{ij} > 0$ if $(j,i) \in \mathcal{E}$, $s_{ij} = 0$ otherwise, and $s_{ii} = 0$. In considered case, the nonzero elements of the adjacency matrix are equal to one. The ability of agents to obtain information about the reference output trajectory is determined by the matrix $\mathcal{R} = \mathrm{diag}[r_i]_{i=1}^{N}$, where $r_i = 1$ if $y_{ref}(p)$ is available to agent i, and $r_i = 0$ otherwise.

In other words, the structure of the network is such that each agent has access to the reference trajectory $y_{ref}(p)$ either directly or through another agent.

The trial-to-trial error on trial k is given by

$$e_i(k,p) = y_{ref}(p) - y_i(k,p). \tag{2}$$

The problem is to find a control algorithm (protocol) $u_i(k,p)$ satisfying the following trial-to-trial convergence and control boundedness conditions:

$$|e_i(k,p)| \leq \kappa\zeta^k, \quad \kappa > 0,\ 0 < \zeta < 1, \tag{3}$$

$$|u_i(k,p)| < \infty, \tag{4}$$

$$1 \leq i \leq N,\ 0 \leq p \leq T-1.$$

According to these conditions, for all agents, the trial-to-trial error decays no slower than a geometric progression as $k \to \infty$ and will remain bounded for all k. Therefore, the tracking accuracy of the desired output trajectory will gradually improve with each trial.

In ILC, once a trial is complete, all generated information is available for updating the control input on the subsequent trial. A common way of using such information is to set

$$u_i(k+1,p) = u_i(k,p) + \Delta u_i(k,p), \tag{5}$$

where $\Delta u_i(k,p)$ is the update term. With such a structure, the problem is reduced to determining the update term under which the ILC law (5) will ensure conditions (3) and (4).

3 Problem Solution

The problem will be solved using an extension of the approach proposed in [Pakshin et al., 2024]. This approach is based on the heavy-ball method, which accelerates the convergence of the gradient method in classical optimization problems [Polyak, 1987].

The dynamics of the network system will be presented in the standard discrete repetitive process form. To do this, the equations of system (1) with the law (5) and the trial-to-trial error (2) are written with respect to the increment of the state vector

$$\eta_i(k,p+1) = x_i(k+1,p) - x_i(k,p)$$

as follows:

$$\begin{aligned} \eta_i(k,p+1) &= A\eta_i(k,p) + B\Delta u_i(k,p-1),\\ e_i(k+1,p) &= -CA\eta_i(k,p) + e_i(k,p)\\ &\quad - CB\Delta u_i(k,p-1). \end{aligned} \tag{6}$$

For a single linear system, the update term is determined by minimizing the squared trial-to-trial error [Pakshin et al., 2024]. In this case, the system always has access to the reference trajectory. In a network system, however, an agent has access either to the reference output trajectory or only to the outputs of its neighbors, depending on the network topology. The heterogeneity of information about the reference output trajectory is taken into account by introducing a weighted deviation of the agent's output from the trajectories available to it:

$$\begin{aligned} d_i(k+1,p) &= r_i\big(y_{ref}(p) - y_i(k+1,p)\big)\\ &\quad + \sum_{j=1}^{N} s_{ij}\big(y_j(k,p) - y_i(k+1,p)\big), \end{aligned}$$

where r_i and s_{ij} are entries of the matrices $\mathcal{R}$ and $\mathcal{S}$, respectively; the corresponding deviation relative to the trial-to-trial error is given by

$$d_i(k+1,p) = \left(r_i + \sum_{j=1}^{N} s_{ij}\right) e_i(k+1,p) - \sum_{j=1}^{N} s_{ij} e_j(k,p). \quad (7)$$

Let $\mathcal{H} = \text{diag}[h_i]_{i=1}^N$, where $h_i = r_i + \sum_{j=1}^N s_{ij}$, i.e., the element h_i represents the sum of the weights of the trajectories available to agent i. As a result, the weighted mean error can be determined from (7) as

$$w_i(k+1,p) = \frac{1}{h_i} d_i(k+1,p) = e_i(k+1,p) - \frac{1}{h_i}\sum_{j=1}^{N} s_{ij} e_j(k,p). \quad (8)$$

The update term will be obtained by minimizing $\big(w_i(k+1,p)\big)^2$ using the heavy ball method. According to this method,

$$\begin{aligned}\Delta u_i(k,p-1) &= \alpha \Delta u_i(k-1,p-1) + \beta \nabla\left(\frac{1}{2}\big(w_i(k+1,p)\big)^2\right) \\ &\quad + \gamma\big(\Delta u_i(k-1,p-1) - \Delta u_i(k-2,p-1)\big), \quad (9)\end{aligned}$$

where

$$\nabla\left(\frac{1}{2}\big(w_i(k+1,p)\big)^2\right) = \frac{\partial}{\partial \Delta u_i(k,p-1)}\left(\frac{1}{2}\big(w_i(k+1,p)\big)^2\right). \quad (10)$$

In view of the last equation in (6), calculating the right-hand side of (10) gives

$$\begin{aligned}\nabla\left(\frac{1}{2}\big(w_i(k+1,p)\big)^2\right) &= CBCA\eta_i(k,p) - CBe_i(k,p) + CB\left(\frac{1}{h_i}\sum_{j=1}^{N} s_{ij} e_j(k,p)\right) \\ &\quad + (CB)^2 \Delta u_i(k,p-1).\end{aligned}$$

With this expression substituted into (9), after routine additional transformations, the update term takes the form

$$\begin{aligned}\Delta u_i(k,p-1) &= K_1 \Delta u_i(k-1,p-1) + K_2 CA\eta_i(k,p) - K_2 e_i(k,p) \\ &\quad + K_2\left(\frac{1}{h_i}\sum_{j=1}^{N} s_{ij} e_j(k,p)\right) - K_3 \Delta u_i(k-2,p-1), \quad (11)\end{aligned}$$

where

$$K_1 = \frac{\alpha+\gamma}{1-\beta(CB)^2}, \quad K_2 = \frac{\beta CB}{1-\beta(CB)^2}, \quad K_3 = \frac{\gamma}{1-\beta(CB)^2}$$

are generalized parameters, in terms of which it is possible to use technique of the linear matrix inequalities (LMI) below.

In terms of the variables of model (1) and the entries of the communication graph matrices, the update term is explicitly written as

$$\begin{aligned}\Delta u_i(k,p-1) &= K_1 \Delta u_i(k-1,p-1) + K_2 CA\big(x_i(k+1,p-1) - x_i(k,p-1)\big) \\ &\quad - K_2\left(\frac{1}{r_i + \sum_{j=1}^N s_{ij}}\left(r_i y_{ref}(p) + \sum_{j=1}^{N} s_{ij} y_j(k,p)\right) - y_i(k,p)\right) \\ &\quad - K_3 \Delta u_i(k-2,p-1). \quad (12)\end{aligned}$$

In the case of agent i as a leader with $r_i = 1$ and $s_{ij} = 0$ for all j, the problem under consideration is reduced to minimizing $\big(e_i(k+1,p)\big)^2$; and substituting the coefficients r_i and s_{ij} into (12) gives an update law similar to that obtained in [Pakshin et al., 2024] for a single linear system. In the case of follower agents with $r_i = 0$, the resulting update law will ensure the convergence of the output to $y_{ref}(p)$ when the output of each agent j with $s_{ij} > 0$ converges to the reference trajectory.

Now the problem is to find appropriate matrices K_1, K_2, and K_3 under which the protocol (5) with the update term (11) will ensure conditions (3) and (4). By denoting $\nu_i(k,p) = \Delta u_i(k-1,p-1)$, one compactly rewrites system (6) with the adopted update law as

$$\begin{aligned}\eta_i(k,p+1) &= (I + BK_2C)A\eta_i(k,p) - BK_2 e_i(k,p) + BK_2\left(\frac{1}{h_i}\sum_{j=1}^{N} s_{ij} e_j(k,p)\right) \\ &\quad + BK_1\nu_i(k,p) - BK_3\mu_i(k,p), \\ e_i(k+1,p) &= -C(I+BK_2C)A\eta_i(k,p) + (1 + CBK_2)e_i(k,p) \\ &\quad - CBK_2\left(\frac{1}{h_i}\sum_{j=1}^{N} s_{ij} e_j(k,p)\right) \\ &\quad - CBK_1\nu_i(k,p) + CBK_3\mu_i(k,p), \\ \nu_i(k+1,p) &= K_2CA\eta_i(k,p) - K_2e_i(k,p) + K_2\left(\frac{1}{h_i}\sum_{j=1}^{N} s_{ij} e_j(k,p)\right) + K_1\nu_i(k,p) \\ &\quad - K_3\mu_i(k,p), \\ \mu_i(k+1,p) &= \nu_i(k,p).\end{aligned} \quad (13)$$

In this model, the variables $e_i(k,p)$, $\nu_i(k,p)$, and $\mu_i(k,p)$ change relative to k. Based on this feature, it is convenient to combine them into the vector

$$\epsilon_i(k,p) = \big[\nu_i(k,p)\ \mu_i(k,p)\ e_i(k,p)\big]^\top$$

and rewrite (13) as follows:

$$\begin{aligned}
\eta_i(k,p+1) &= (A_{11} + B_1KC_1)\eta_i(k,p) \\
&\quad + (A_{12} + B_1KC_{21})\epsilon_i(k,p) \\
&\quad + B_1KC_{22}\bigg(\frac{1}{h_i}\sum_{j=1}^{N} s_{ij}\epsilon_j(k,p)\bigg), \\
\epsilon_i(k+1,p) &= (A_{21} + B_2KC_1)\eta_i(k,p) \\
&\quad + (A_{22} + B_2KC_{21})\epsilon_i(k,p) \\
&\quad + B_2KC_{22}\bigg(\frac{1}{h_i}\sum_{j=1}^{N} s_{ij}\epsilon_j(k,p)\bigg),
\end{aligned} \tag{14}$$

where

$$A_{11} = A, \quad A_{12} = 0, \quad A_{21} = \begin{bmatrix} 0 \\ 0 \\ -CA \end{bmatrix},$$

$$A_{22} = \begin{bmatrix} 0 & 0 & 0 \\ I & 0 & 0 \\ 0 & 0 & I \end{bmatrix},\ B_1 = B, \quad B_2 = \begin{bmatrix} I \\ 0 \\ -CB \end{bmatrix},$$

$$C_1 = \begin{bmatrix} 0 \\ CA \\ 0 \end{bmatrix},\ C_{21} = \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & -I \\ 0 & -I & 0 \end{bmatrix},$$

$$C_{22} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{bmatrix},\ K = \big[K_1\ K_2\ K_3\big].$$

With the extended vectors

$$\eta(k,p) = \Big[\eta_1(k,p)^\top\ \dots\ \eta_N(k,p)^\top\Big]^\top,$$
$$\epsilon(k,p) = \Big[\epsilon_1(k,p)^\top\ \dots\ \epsilon_N(k,p)^\top\Big]^\top,$$

the network system dynamics can be finally modeled in the standard discrete repetitive process form:

$$\begin{aligned}
\eta(k,p+1) &= (\bar{A}_{11} + \bar{B}_1\bar{K}\bar{C}_1)\eta(k,p) \\
&\quad + (\bar{A}_{12} + \bar{B}_1\bar{K}\bar{C}_2)\epsilon(k,p), \\
\epsilon(k+1,p) &= (\bar{A}_{21} + \bar{B}_2\bar{K}\bar{C}_1)\eta(k,p) \\
&\quad + (\bar{A}_{22} + \bar{B}_2\bar{K}\bar{C}_2)\epsilon(k,p),
\end{aligned} \tag{15}$$

where

$$\bar{A}_{11} = I_N \otimes A_{11}, \quad \bar{A}_{12} = I_N \otimes A_{12},$$
$$\bar{A}_{21} = I_N \otimes A_{21},\ \bar{A}_{22} = I_N \otimes A_{22}, \quad \bar{B}_1 = I_N \otimes B_1,$$
$$\bar{B}_2 = I_N \otimes B_2,\ \bar{C}_1 = I_N \otimes C_1,$$
$$\bar{C}_2 = (I_N \otimes C_{21}) + (\mathcal{H}^{-1}\mathcal{S} \otimes C_{22}),\ \bar{K} = I_N \otimes K$$

and the symbol $\otimes$ means the Kronecker product.

A convergence condition of the trial-to-trial error will be established using the method of vector Lyapunov functions for repetitive processes [Pakshin et al., 2016]. Consider a vector function of the form

$$V\big(\eta(k,p),\epsilon(k,p)\big) = \begin{bmatrix} V_1\big(\eta(k,p)\big) \\ V_2\big(\epsilon(k,p)\big) \end{bmatrix}, \tag{16}$$

where $V_1\big(\eta(k,p)\big) > 0$, $\eta(k,p) \neq 0$, $V_2\big(\epsilon(k,p)\big) > 0$, $\epsilon(k,p) \neq 0$, $V_1(0) = 0$, and $V_2(0) = 0$, and define the counterpart of its divergence as

$$\begin{aligned}
&\mathcal{D}V\big(\eta(k,p),\epsilon(k,p)\big) = V_1\big(\eta(k,p+1)\big) \\
&- V_1\big(\eta(k,p)\big) + V_2\big(\epsilon(k+1,p)\big) - V_2\big(\epsilon(k,p)\big).
\end{aligned} \tag{17}$$

Then the following theorem from [Pakshin et al., 2016] is applicable.

Theorem 1. *Let there exist a vector Lyapunov function* (16) *and positive scalars c_1, c_2, and c_3 such that the following inequalities hold along the trajectories of the system* (15)*:*

$$c_1\|\eta(k,p)\|^2 \le V_1\big(\eta(k,p)\big) \le c_2\|\eta(k,p)\|^2,$$
$$c_1\|\epsilon(k,p)\|^2 \le V_2\big(\epsilon(k,p)\big) \le c_2\|\epsilon(k,p)\|^2,$$
$$\mathcal{D}V\big(\eta(k,p),\epsilon(k,p)\big) \le -c_3\big(\|\eta(k,p)\|^2 + \|\epsilon(k,p)\|^2\big).$$

Then the iterative learning control algorithm (5) *with the update law* (11) *ensures the trial-to-trial error convergence* (3) *and control boundedness* (4) *conditions.*

Let the entries of the vector Lyapunov function (16) be chosen as quadratic forms

$$\begin{aligned}
V_1\big(\eta(k,p)\big) &= \eta(k,p)^\top \bar{P}_1\eta(k,p), \\
V_2\big(\epsilon(k,p)\big) &= \epsilon(k,p)^\top \bar{P}_2\epsilon(k,p),
\end{aligned} \tag{18}$$

where $\bar{P}_1 = I_N \otimes P_1$ and $\bar{P}_2 = I_N \otimes P_2$. Then, by calculating the divergence (17) along the trajectories of system (15), one satisfies the conditions of Theorem 1 if, for some matrices $Q_1 \succ 0$ and $Q_2 \succ 0$ of dimensions $n_x \times n_x$ and 3×3, respectively, and a scalar $\rho > 0$,

$$\begin{aligned}
&\big(\bar{A} + \bar{B}\bar{K}\bar{C}\big)^\top\bar{P}\big(\bar{A} + \bar{B}\bar{K}\bar{C}\big) - \bar{P} + \bar{Q} \\
&\qquad + (\bar{K}\bar{C})^\top\bar{R}\bar{K}\bar{C} \preceq 0,\ \bar{P} \succ 0,
\end{aligned} \tag{19}$$

where

$$\bar{A} = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} \bar{B}_1 \\ \bar{B}_2 \end{bmatrix}, \quad \bar{C} = \big[\bar{C}_1\ \bar{C}_2\big],$$
$$\bar{P} = \operatorname{diag}\big[\bar{P}_1\ \bar{P}_2\big],\ \bar{Q} = \operatorname{diag}\big[\bar{Q}_1\ \bar{Q}_2\big], \quad \bar{Q}_1 = I_N \otimes Q_1,$$
$$\bar{Q}_2 = I_N \otimes Q_2, \quad \bar{R} = I_N \otimes \rho.$$

Based on the Schur complement lemma, inequalities (19) can be written as a system of matrix equations and inequalities:

$$\begin{bmatrix} \bar{X} & M_{12} & \bar{X} & (\bar{Y}\bar{C})^\top \\ M_{12}^\top & \bar{X} & 0 & 0 \\ \bar{X} & 0 & \bar{Q}^{-1} & 0 \\ \bar{Y}\bar{C} & 0 & 0 & \bar{R}^{-1} \end{bmatrix} \succeq 0,$$
$$\bar{X} \succ 0, \quad \bar{C}\bar{X} = \bar{Z}\bar{C}, \quad (20)$$

where

$$M_{12} = \left(\bar{A}\bar{X} + \bar{B}\bar{Y}\bar{C}\right)^\top,$$
$$\bar{X} = \text{diag}\left[\bar{X}_1 \ \bar{X}_2\right] = P^{-1}, \quad \bar{X}_1 = I_N \otimes X_1,$$
$$\bar{X}_2 = I_N \otimes X_2, \ \bar{Y} = I_N \otimes Y,$$
$$Y = KZ, \quad \bar{Z} = I_N \otimes Z,$$

and X_1, X_2, Y, and Z are the desired matrices of dimensions $n_x \times n_x$, 3×3, 1×3, and 3×3, respectively. If $\bar{C}$ has full row rank, then the equation from (20) has unique solution relative to Z and Z is also has full rank, and hence invertible [Crusius and Trofino, 1999]. The above results are summarized as follows.

Theorem 2. *Suppose that matrix $\bar{C}$ has full row rank and for some matrices $Q_1 \succ 0$, $Q_2 \succ 0$, and a scalar $\rho > 0$, the system of matrix equations and inequalities* (20) *has a solution X_1, X_2, Y, and Z. Then the ILC law* (5) *with the update term* (12) *and the coefficients*

$$K = \left[K_1 \ K_2 \ K_3\right] = YZ^{-1}$$

ensures the trial-to-trial error convergence (3) *and control boundedness* (4) *conditions.*

Thus, a control algorithm satisfying Theorem 2 ensures that the output's convergence to the reference trajectory. Here, Q_1, Q_2, and ρ play a role similar to the weight matrices in the LQR theory; they are selected based on knowledge of the system dynamics and its desired performance.

4 Example

As an example, consider a network system consisting of three simple single-link gantry robots (agents). Each agent transfers payloads along a given trajectory in the horizontal plane in a repetitive mode. The agent's dynamics model has the form

$$J\ddot{\theta}(t) + K_s\theta(t) = u(t), \quad 0 \le t \le T_c,$$

where $\theta(t)$ is the link rotation angle; $J = m_l l^2 + \frac{1}{3}m_p l^2$ is the moment of inertia; m_p is the mass of the payload; m_l is the mass of the rigid link; l is the link length; K_s is the stiffness coefficient; $u(t)$ is the control input; and $T_c < \infty$ is the trial length. The continuous-time state-space model with respect to the vectors $x_i(k,t) = [\theta \ \dot{\theta}]^\top$ and $y_i(k,t) = \dot{\theta}$ can be written as

$$\begin{aligned} \dot{x}_i(k,t) &= A_c x_i(k,t) + B_c u_i(k,t), \\ y_i(k,t) &= C_c x_i(k,t), \end{aligned} \quad (21)$$

where

$$A_c = \begin{bmatrix} 0 & 1 \\ -K_s/J & 0 \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ 1/J \end{bmatrix}, \quad C_c = \begin{bmatrix} 0 \ 1 \end{bmatrix}.$$

Let the parameter values be $m_p = 5$ kg, $m_l = 1$ kg, $l = 0.3$ m, and $K_s = 2.3$. The chosen reference trajectory of the angular velocity,

$$\dot{\theta}_{ref}(t) = \frac{1}{4}\left(\pi t - \frac{\pi t^2}{2}\right),$$

specifies a turn of 30° in $T_c = 2$ s along the trajectory

$$\theta_{ref}(t) = \frac{1}{4}\left(\frac{\pi t^2}{2} - \frac{\pi t^3}{6}\right).$$

The agents are connected serially, with the first agent as the leader. This network configuration is defined by the matrices

$$\mathcal{S} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathcal{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathcal{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The standard MATLAB functions `ss` and `c2d` were used to construct the discrete model. The sampling step was taken to be 10^{-2} s, so $T = 200$. For comparative analysis, the networked ILC algorithm with the update term

$$\begin{aligned} \Delta u_i(k,p-1) &= \mathbf{K}_1\big(x_i(k+1,p-1) \\ &- x_i(k,p-1)\big) + \mathbf{K}_2\Big(r_i\big(y_{ref}(p) - y_i(k,p)\big) \\ &+ \sum_{j=1}^{N} s_{ij}\big(y_j(k,p) - y_i(k,p)\big)\Big) \end{aligned} \quad (22)$$

was applied (for example, see [Pakshin et al., 2018]). Similarly, by Theorem 1, the ILC law (5) with this update term ensures conditions (3) and (4) if the system

$$\begin{bmatrix} \bar{\mathbf{X}} & \mathbf{M}_{12} & \bar{\mathbf{X}} & (\bar{\mathbf{Y}}\bar{\mathbf{C}})^\top & \\ \mathbf{M}_{12}^\top & \bar{\mathbf{C}} & \bar{\mathbf{X}} & 0 & 0 \\ \bar{\mathbf{X}} & 0 & \bar{\mathbf{Q}}^{-1} & 0 & \\ \bar{\mathbf{Y}}\bar{\mathbf{C}} & 0 & 0 & \bar{\mathbf{R}}^{-1} & \end{bmatrix} \succeq 0, \ \bar{\mathbf{X}} \succ 0, \quad (23)$$

where

$$\mathbf{M}_{12} = \left(\bar{\mathbf{A}}\bar{\mathbf{X}} + \bar{\mathbf{B}}\bar{\mathbf{Y}}\bar{\mathbf{C}}\right)^{\top},$$
$$\bar{\mathbf{A}} = \begin{bmatrix} I_N \otimes A & 0 \\ I_N \otimes (-CA) & I_N \end{bmatrix}, \ \bar{\mathbf{B}} = \begin{bmatrix} I_N \otimes B \\ I_N \otimes (-CB) \end{bmatrix},$$
$$\bar{\mathbf{C}} = \mathrm{diag}\left[I_{Nn_x} \ \mathcal{L} + \mathcal{R}\right],$$
$$\bar{\mathbf{X}} = \mathrm{diag}\left[I_N \otimes \mathbf{X}_1 \ I_N \otimes \mathbf{X}_2\right],$$
$$\bar{\mathbf{Y}} = \left[I_N \otimes \mathbf{Y}_1 \ I_N \otimes \mathbf{Y}_2\right],$$
$$\bar{\mathbf{Q}} = \mathrm{diag}\left[I_N \otimes \mathbf{Q}_1 \ I_N \otimes \mathbf{Q}_2\right], \ \bar{\mathbf{R}} = I_N \otimes \mathbf{R},$$

is solvable with respect to $\mathbf{X}_1$, $\mathbf{X}_2$, $\mathbf{Y}_1$, and $\mathbf{Y}_2$ for some matrices $\mathbf{Q}_1 \succ 0$, $\mathbf{Q}_2 \succ 0$ and scalar $\mathbf{R} > 0$, while

$$\mathbf{K}_1 = \mathbf{Y}_1\mathbf{X}_1^{-1}, \quad \mathbf{K}_2 = \mathbf{Y}_2\mathbf{X}_2^{-1}.$$

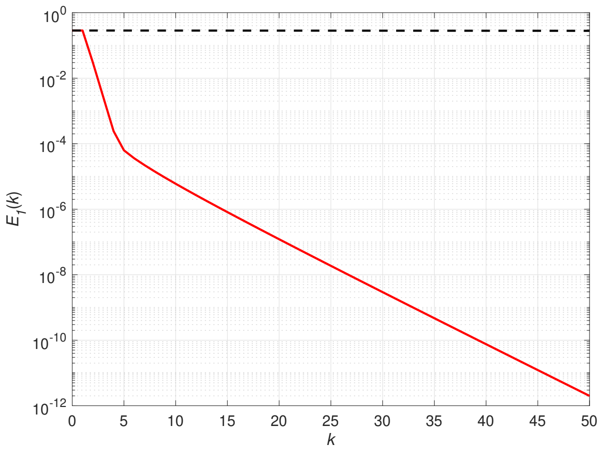

Figure 1. The leader's progression of $E(k)$ under the ILC law with the update term (22) (dashed lines) and with the update term (11) (solid lines) with identity matrices $\bar{Q}$, $\bar{R}$, $\bar{\mathbf{Q}}$ and $\bar{\mathbf{R}}$.

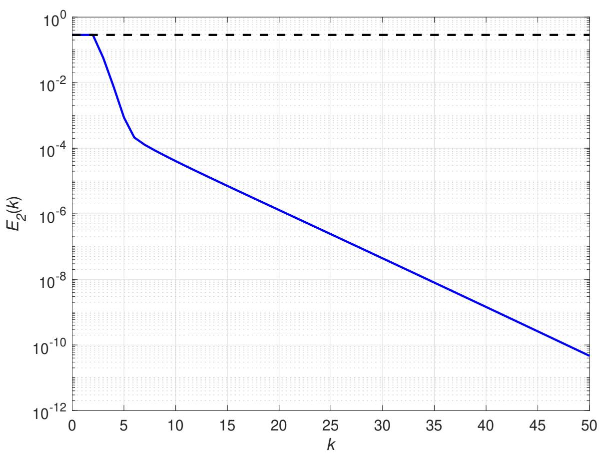

Figure 2. The first follower's progression of $E(k)$ under the ILC law with the update term (22) (dashed lines) and with the update term (11) (solid lines) with identity matrices $\bar{Q}$, $\bar{R}$, $\bar{\mathbf{Q}}$ and $\bar{\mathbf{R}}$.

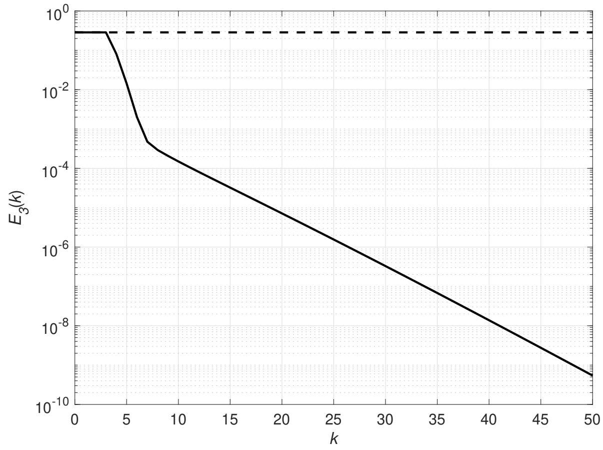

Figure 3. The second follower's progression of $E(k)$ under the ILC law with the update term (22) (dashed lines) and with the update term (11) (solid lines) with identity matrices $\bar{Q}$, $\bar{R}$, $\bar{\mathbf{Q}}$ and $\bar{\mathbf{R}}$.

Here, $\mathcal{L} = [l_{ij}]_{i,j=1}^{N}$ is the Laplace matrix in which $l_{ij} = -s_{ij}$ if $i \neq j$, and $l_{ii} = \sum_{n=1}^{N} s_{in}$.

To assess the performance of the ILC algorithm, let us introduce the root-mean-square (RMS) trial-to-trial error

$$E_i(k) = \sqrt{\frac{1}{T}\sum_{p=0}^{T-1} |e_i(k,p)|^2} \tag{24}$$

as a measure of the convergence of the trial-to-trial error over trials.

For Q_1, Q_2 as identity matrices and $\rho = 1$, computations based on Theorem 2 yielded

$$K_1 = -5.9426 \times 10^{-8}, \quad K_2 = -11.069,$$
$$K_3 = 9.0801 \times 10^{-9}.$$

For the ILC algorithm (22) and the same weights, solving (23) gave

$$\mathbf{K}_1 = \left[2.2947 \ -23.9923\right], \quad \mathbf{K}_2 = 0.0104.$$

The ILC law with the update tem (11) reduced in one trial the leader's RMS trial-to-trial error by 10 times, and the followers' one by more than 3 times (from the start of the learning process, which occurs with a delay of one trial relative to the leader). On trial 7, the agents' trial-to-trial errors reached values below 10^{-3}. For comparison, under the ILC algorithm with the update law (22), agents fail to achieve such accuracy in 50 trials: their trial-to-trial errors during that time decreased by no more than 2.2% (figures 1 – 3).

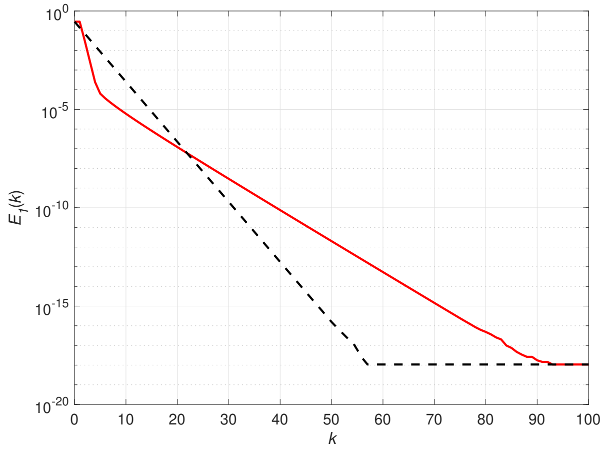

Figure 4. The leader's progression of $E(k)$ under the ILC algorithm with the update law (22) and $\mathbf{Q}_1$, $\mathbf{Q}_2$, $\mathbf{R}$ given by (25) (dashed lines) and with the update term (11), $\bar{Q} = I$, and $\bar{R} = I$ (solid lines).

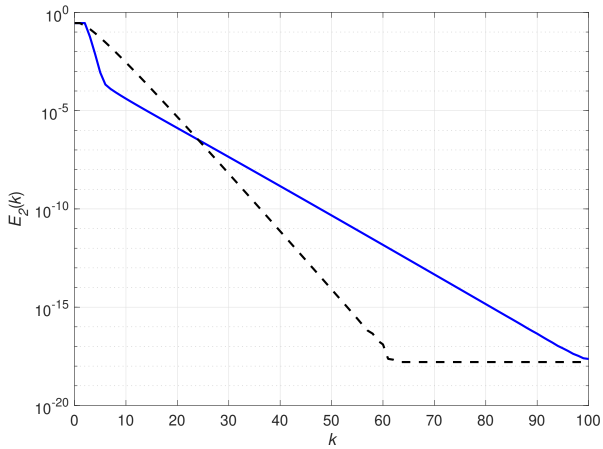

Figure 5. The first follower's progression of $E(k)$ under the ILC law with the update term (22) and $\mathbf{Q}_1$, $\mathbf{Q}_2$, $\mathbf{R}$ given by (25) (dashed lines) and with the update term (11), $\bar{Q} = I$, and $\bar{R} = I$ (solid lines).

For the ILC law with the update term (22), similar convergence characteristics can be achieved through a heuristic selection of $\mathbf{Q}_1$, $\mathbf{Q}_2$, and $\mathbf{R}$, but this is often a challenge. In this case, the choice

$$\mathbf{Q}_1 = 10^{-6} I_{n_x}, \quad \mathbf{Q}_2 = 10^5, \quad \mathbf{R} = 10, \tag{25}$$

allowed achieving a tracking accuracy of 10^{-3} for all agents in 15 trials. This is twice as slow as with the update law (11), but this law faster reaches values near computer zero (see Figs. 4– 6).

5 Conclusions

The new ILC design method proposed in this paper ensures high convergence rate and does not require heuristic parameter selection. However, as shown in [Koposov and Pakshin, 2023], random disturbances acting on the plant and measurement noise can significantly reduce achievable accuracy. Therefore, in the near future, it is planned to extend the above approach to the case of these factors.

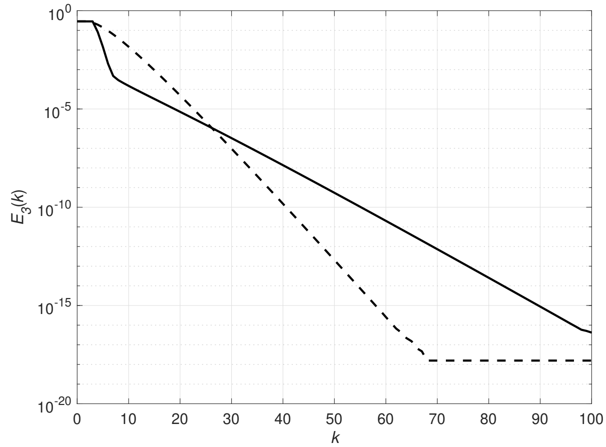

Figure 6. The second agent's progression of $E(k)$ under the ILC law with the update term (22) and $\mathbf{Q}_1$, $\mathbf{Q}_2$, $\mathbf{R}$ given by (25) (dashed lines) and with the update term (11), $\bar{Q} = I$, and $\bar{R} = I$ (solid lines).

6 Acknowledgement

This work was supported by the Russian Science Foundation under grant 25-11-00065, https://rscf.ru/en/project/25-11-00065/.

References

Ahn, H.-S., Chen, Y.-Q., and Moore, K. L. (2007). Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, **37** (6), pp. 1099–1121.

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

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

Crusius, C. and Trofino, A. (1999). Sufficient lmi conditions for output feedback control problems. *IEEE Transactions on Automatic Control*, **44** (5), pp. 1053–1057.

Freeman, C. T., Rogers, E., Hughes, A.-M., Burridge, J. H., and Meadmore, K. L. (2012). Iterative learning control in health care: electrical stimulation and robotic-assisted upper-limb stroke rehabilitation. *IEEE Control Systems Magazine*, **32** (1), pp. 18–43.

Gu, P., Tian, S., and Chen, Y. (2019). Iterative learning control based on Nesterov accelerated gradient method. *IEEE Access*, **7**, pp. 115836–115842.

Hock, A. and Schoellig, A. P. (2019). Distributed iterative learning control for multi-agent systems. *Autonomous Robots*, **43**, pp. 1989–2010.

Ketelhut, M., Stemmler, S., Gesenhues, J., Hein, M., and Abel, D. (2019). Iterative learning control of ventricular assist devices with variable cycle durations. *Control Engineering Practice*, **83**, pp. 33–44.

Konovalov, P. and Matveev, A. (2025). Distributed algorithms for self-spreading of robotic networks over unknown complex areas in gps-defined environments. *Cybernetics and Physics*, **14** (1), pp. 52–61.

Koposov, A. S. and Pakshin, P. V. (2023). Iterative learning control of stochastic multi-agent systems with variable reference trajectory and topology. *Automation and Remote Control*, **84** (6), pp. 612–625.

Lim, I., Hoelzle, D. J., and Barton, K. L. (2017). A multi-objective iterative learning control approach for additive manufacturing applications. *Control Engineering Practice*, **64**, pp. 74–87.

Liu, Y. and Jia, Y. (2012). An iterative learning approach to formation control of multi-agent systems. *Systems & Control Letters*, **61**, pp. 148–154.

Meadmore, K. L., Exell, T. A., Hallewell, E., Hughes, A.-M., Freeman, C. T., Kutlu, M., Benson, V., Rogers, E., and Burridge, J. H. (2014). The application of precisely controlled functional electrical stimulation to the shoulder, elbow and wrist for upper limb stroke rehabilitation: a feasibility study. *Journal of NeuroEngineering and Rehabilitation*, **11** (art. no. 105).

Meng, D., Du, W., and Jia, Y. (2015). Data-driven consensus control for networked agents: an iterative learning control-motivated approach. *IET Control Theory & Applications*, **9**, pp. 2084–2096.

Pakshin, P., Emelianova, J., Emelianov, M., Gałkowski, K., and Rogers, E. (2016). Dissipivity and stabilization of nonlinear repetitive processes. *Systems & Control Letters*, **91**, pp. 14–20.

Pakshin, P., Emelianova, J., and Rogers, E. (2024). State observer-based iterative learning control design for discrete systems using the heavy ball method. *Autom. Remote Control*, **85**, pp. 727–740.

Pakshin, P. V., Emelianova, J. P., and Emelianov, M. A. (2018). Iterative learning control design for multiagent systems based on 2D models. *Automation and Remote Control*, **79** (6), pp. 1040–1056.

Polyak, B. T. (1987). *Introduction to optimization*. Optimization Software, New York.

Rogers, E., Chu, B., Freeman, C., and Lewin, P. (2023). *Iterative Learning Control Algorithms and Experimental Benchmarking*. John Wiley & Sons, Chichester, UK.

Sammons, P. M., Gegel, M. L., Bristow, D. A., and Landers, R. G. (2019). Repetitive process control of additive manufacturing with application to laser metal deposition. *IEEE Transactions on Control Systems Technology*, **27** (2), pp. 566–577.

Sharma, A. and Lather, J. S. (2024). Synchronization of coupled quadcopters using contraction theory. *Cybernetics and Physics*, **13** (2), pp. 142–151.

Yu, X., Hou, Z., and Polycarpou, M. M. (2022). Distributed data-driven iterative learning consensus tracking for nonlinear discrete-time multiagent systems. *IEEE Transactions on Automatic Control*, **67**, pp. 3670–3677.

Zhang, Z., Jiang, H., and Shen, D. (2024). Fast iterative learning control algorithms based on heavy ball with adaptive stepsize. In *43rd Chinese Control Conference (CCC)*, Kunming, China, July, pp. 2552– 2557.