SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2024, Vol. 67, Iss. 3, 132203:1–132203:12
https://doi.org/10.1007/s11432-023-3845-6

Segment-wise learning control for trajectory tracking of robot manipulators under iteration-dependent periods

Fan ZHANG[1,2], Deyuan MENG[1,2*] & Kaiquan CAI[3]

[1] *The Seventh Research Division, Beihang University, Beijing* 100191*, China;*
[2] *School of Automation Science and Electrical Engineering, Beihang University, Beijing* 100191*, China;*
[3] *School of Electronics and Information Engineering, Beihang University, Beijing* 100191*, China*

Received 17 April 2023/Revised 27 June 2023/Accepted 25 July 2023/Published online 20 February 2024

Abstract This paper is concerned with the amplitude boundedness problem of adaptive iterative learning control (AILC) for robot manipulators operating with iteration-dependent periods. By introducing virtual memory slots for storing historical data, a practical AILC method is proposed to achieve the segment-wise learning. This method requires less memory storage for historical information of previous iterations, especially in comparison with that of the conventional AILC methods using point-wise learning strategies. It is shown that not only the energy boundedness but also the amplitude boundedness of estimates and inputs of practical AILC can be guaranteed. Moreover, the practical AILC method can achieve the perfect tracking objective regardless of iteration-dependent periods when the robot manipulators have a persistent full learning property. In addition, a solution to the visual manipulator platform is provided and deployed based on Coppeliasim and Matlab, which helps to show the amplitude boundedness of learning results and the perfect tracking performances of the proposed practical AILC method for robot manipulators.

Keywords amplitude boundedness, iteration-dependent period, iterative learning control, robot manipulator, segment-wise, virtual memory slot

1 Introduction

Iterative learning control (ILC) is generally considered one of the intelligence approaches for rigid robot manipulators and other practical applications which repeatedly operate within a fixed and finite period [1–5]. Compared with existing feedback control methods, such as PD control and sliding mode control [6, 7], ILC explicitly shows greater advantages in improving the transient performance and achieving the perfect tracking for some desired outputs [8]. Specifically, ILC stores historical information from previous iterations and utilizes it to refine the tracking performance of subsequent iterations [9]. In the field of ILC, most efforts have been devoted to developing controllers under two main frameworks: contraction mapping and composite energy function [10,11]. In contrast with contraction mapping-based ILC, composite energy function-based ILC, also called adaptive ILC (AILC), can achieve the perfect tracking objective even when iteration-dependent uncertainties exist. However, certain issues, such as the amplitude boundedness problem and the large memory occupation of historical data, have been hindering the further applications of AILC, especially when periods are iteration-dependent.

The iteration-independent period condition cannot always be guaranteed in many practical applications due to unknown uncertainties or unpredictable factors [11,12]. For such problems, various AILC methods have been reported in some studies, such as [11, 13]. Generally, a virtual error mechanism is designed to compensate for the missed error information and update the estimations within the non-operation periods. However, the continuity of the estimations along the time axis may be lost, which can be observed according to (6) in [11] or (23) in [13]. This discontinuity further causes the amplitude boundedness problem, which refers to the $\mathscr{L}_{2e}$-norm boundedness of estimates and inputs being guaranteed, but not

* Corresponding author (email: dymeng@buaa.edu.cn)

their $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness. This is because the $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness of continuous functions can be naturally deduced if they are $\mathscr{L}_{2\mathrm{e}}$-norm bounded, but unfortunately, there is no such conclusion for discontinuous functions. The introduction of two kinds of boundedness properties can be found at the end of this section. Note that a similar issue is discussed in a comment and its reply [14–16]. Some studies claim that the prior information can be used to provide the $\mathscr{L}_{\infty e}$ boundedness, but it conflicts with the assumption of uncertainties with unknown bounds.

Another interesting issue is the large memory storage required for AILC of continuous systems. In most studies of conventional AILC, a point-wise learning mechanism is generally applied [17–20]. This means that the historical tracking information within the full period of the previous iteration or iterations should be stored and used to learn the desired inputs. A natural thought coming to mind is whether many points can share the same historical data. The feasibility of this idea is illustrated implicitly in [21, 22]. However, this method makes all estimates within the full period approach some identical value, which may lead to excessive control inputs and even exacerbate the chattering phenomenon at some time instants. The question is how to reduce the required historical data while ensuring perfect learning.

Motivated by the above observations, we develop a practical AILC for the tracking control of robot manipulators operating within iteration-dependent periods. In contrast to existing studies, the main contributions of this paper are summarized as follows.

(1) A novel $\mathscr{L}_{\infty\mathrm{e}}$-norm bounded AILC scheme is developed for the perfect tracking of robot manipulators subjected to iteration-dependent periods and other uncertainties, including external disturbances and unmodeled dynamics. A novel analytical method different from composite energy functions (CEF) is also employed to rigorously verify the effectiveness of the proposed AILC. Compared with conventional point-wise AILC [13, 18, 23], this work provides new insights into the real applications of AILC from the perspective of boundedness.

(2) To address the amplitude boundedness problem of the conventional AILC (see [14]), a novel virtual memory slots (VMSs) mechanism is developed to achieve the segment-wise learning. In addition, by storing historical estimates of some specific moments rather than the whole period, the memory storage pressure for the conventional AILC is relieved. Moreover, a trustworthy simulation platform based on Coppeliasim Edu is constructed to verify the effectiveness of the proposed AILC.

The rest of the paper is organized as follows. In Section 2, some preliminary knowledge of robot manipulators with the Euler-Lagrange equations is introduced. In Section 3, the problem is described, and necessary assumptions are stated. In Section 4, the design and analysis of the $\mathscr{L}_{\infty\mathrm{e}}$-norm bounded AILC are proposed. Simulations and conclusion are presented in Sections 5 and 6, respectively.

Notations. Let $\mathbb{Z}_+ = \{0, 1, 2, \ldots\}$ be the set of nonnegative integers, and $\mathbb{Z}_m = \{1, 2, \ldots, m\}$ be the set of integers from 1 to $m \geqslant 1$. For any $A = [a_{ij}] \in \mathbb{R}^{m\times n}$, $\|A\|$ denotes the spectral norm of A, $\|A\|_1$ denotes the maximum column sum matrix norm of A, and $A^{(i,j)}$ denotes the element in row i and column j of A. For any positive definite matrix $B = [b_{ij}] \in \mathbb{R}^{m\times m}$, $\lambda_{\min}(B)$ denotes the minimal eigenvalue. For any $z = [z_i] \in \mathbb{R}^3$, let $\mathrm{sgn}(z) = [\mathrm{sgn}(z_i)] \in \mathbb{R}^3$ denote the sign operator. For any $x(t) \in \mathbb{R}^n$, the $\mathscr{L}_{\mathrm{pe}}$-norm $\|x(t)\|_{\mathrm{pe}}$ is defined as

$$\|x(t)\|_{\mathrm{pe}} \triangleq \begin{cases} \left(\int_0^T \|x(t)\|^p \,\mathrm{d}t\right)^{\frac{1}{p}}, & \text{if } p \in [1, \infty), \\ \sup_{0\leqslant t\leqslant T} \|x(t)\|, & \text{if } p = \infty, \end{cases} \tag{1}$$

where $T \in \mathbb{R}$ is some positive scalar. Further, if $x(t)$ satisfies $\|x(t)\|_{\mathrm{pe}} < \infty$, then it is said to be $\mathscr{L}_{\mathrm{pe}}$-norm bounded. Additionally, the $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness (respectively, the $\mathscr{L}_{2\mathrm{e}}$-norm boundedness) of $x(t)$ is called the amplitude boundedness (respectively, the energy boundedness) of it.

2 Preliminaries

Based on [24], it is stated that n-degree-of-freedom (n-DOF) rigid robot manipulators with unmodeled dynamics and external disturbances can be governed by

$$M(q_k(t))\ddot{q}_k(t) = -C(q_k(t), \dot{q}_k(t))\dot{q}_k(t) - G(q_k(t)) + u_k(t) + d_k(t), \tag{2}$$

where $k \in \mathbb{Z}_+$ is the iteration index; $t \in [0, T_k]$ is the operation period with iteration-dependent terminal time $T_k \in \mathbb{R}$; $T_L \in \mathbb{R}$ and $T_H \in \mathbb{R}$ are the finite least and greatest values of T_k, i.e., $0 < T_L \leqslant T_k \leqslant$

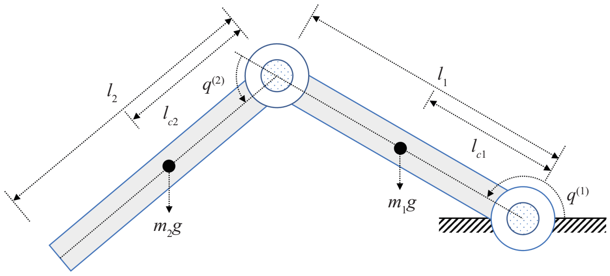

Figure 1 (Color online) Two-DOF rigid robot manipulator. Circles with dotted pattern fill: Joints; Rectangles with gray solid fill: Links; Circles with black solid fill: Centers of mass of links.

$T_H < \infty$, $\forall k \in \mathbb{Z}_+$; $q_k(t) \in \mathbb{R}^n$, $\dot{q}_k(t) \in \mathbb{R}^n$, and $\ddot{q}_k(t) \in \mathbb{R}^n$ are the vectors of joint angle, joint angular velocity, and joint angular acceleration, respectively; $M(q_k(t)) \in \mathbb{R}^{n\times n}$ is the positive definite inertia matrix; $C\left(q_k(t), \dot{q}_k(t)\right) \in \mathbb{R}^{n\times n}$ and $G\left(q_k(t)\right) \in \mathbb{R}^n$ originate from Coriolis, centrifugal, and gravitational torque; $u_k(t) \in \mathbb{R}^n$ and $d_k(t) \in \mathbb{R}^n$ are the control input and random external disturbance, respectively. Note that $M(q_k(t))$, $C(q_k(t), \dot{q}_k(t))$, $G(q_k(t))$, and $d_k(t)$ are unknown, which means that they cannot appear in the controller design. Furthermore, as introduced in [18, 25], the following inequalities hold:

$$\|M(q_k(t))\| \leqslant \beta_M(t), \quad \|C(q_k(t), \dot{q}_k(t))\| \leqslant \beta_C(t)\|\dot{q}_k(t)\|, \quad \|G(q_k(t))\| \leqslant \beta_G(t), \quad \|d_k(t)\| \leqslant \beta_D(t), \tag{3}$$

where $\beta_M(t) \in \mathbb{R}$, $\beta_C(t) \in \mathbb{R}$, $\beta_G(t) \in \mathbb{R}$, and $\beta_D(t) \in \mathbb{R}$ are some finite but unknown bounds. Note that this fact does not require the assumption of the boundedness of $q_k(t)$ because $q_k(t)$ appears in $M(q_k(t))$, $C(q_k(t), \dot{q}_k(t))$, and $G(q_k(t))$ as the trigonometric functions actually. Another well-known property is that $\dot{M}\left(q_k(t)\right) - 2C\left(q_k(t), \dot{q}_k(t)\right)$ is skew symmetric such that

$$x^{\mathrm{T}}\left(\frac{1}{2}\dot{M}\left(q_k(t)\right) - C\left(q_k(t), \dot{q}_k(t)\right)\right)x = 0, \forall x \in \mathbb{R}^n. \tag{4}$$

To facilitate understanding of the reasonableness of the concepts or properties mentioned above, such as (3) and (4), we provide a practical example of a two-DOF rigid robot manipulator, which will also be considered in simulations of Section 5.

Example 1. A planar elbow manipulator with two revolute joints is illustrated in Figure 1. According to Subsection 6.4 of [26], notations can be fixed as follows: for $i = 1, 2$, l_i denotes the length of links i, l_{ci} denotes the distance from the previous joint to the center of mass of link i, $q_k^{(i)}(t)$ denotes the angle of joint i, m_i denotes the mass of link i, g denotes the acceleration due to gravity, I_i denotes the moment of inertia of link i about an axis coming out of the page, passing through the center of mass of link i. Here, the matrix $M(q_k(t))$ can be presented as

$$\begin{aligned}
M^{(1,1)}(q_k(t)) &= m_2\left(l_1^2 + l_{c2}^2 + 2l_1l_{c2}\cos(q_k^{(2)}(t))\right) + m_1l_{c1}^2 + I_1 + I_2,\\
M^{(1,2)}(q_k(t)) &= m_2\left(l_{c2}^2 + l_1l_{c2}\cos(q_k^{(2)}(t))\right) + I_2,\\
M^{(2,1)}(q_k(t)) &= m_2\left(l_{c2}^2 + l_1l_{c2}\cos(q_k^{(2)}(t))\right) + I_2,\\
M^{(2,2)}(q_k(t)) &= m_2l_{c2}^2 + I_2,
\end{aligned}$$

the vector $G\left(q_k(t)\right)$ is given by

$$\begin{aligned}
G^{(1)}\left(q_k(t)\right) &= \left(m_1l_{c1} + m_2l_1\right)g\cos(q_k^{(1)}(t)) + m_2l_{c2}g\cos(q_k^{(1)}(t) + q_k^{(2)}(t)),\\
G^{(2)}\left(q_k(t)\right) &= m_2l_{c2}g\cos(q_k^{(1)}(t) + q_k^{(2)}(t)),
\end{aligned}$$

and matrix $C\left(q_k(t), \dot{q}_k(t)\right)$ is given by

$$C\left(q_k(t), \dot{q}_k(t)\right) = m_2l_1l_{c2}\sin(q_k^{(2)}(t))\begin{bmatrix} -\dot{q}_k^{(2)}(t) & -\dot{q}_k^{(1)}(t) - \dot{q}_k^{(2)}(t) \\ \dot{q}_k^{(1)}(t) & 0 \end{bmatrix}.$$

3 Problem formulation

Before presenting the detailed objective, we need to restate the amplitude boundedness problem in the conventional point-wise AILC for applications operating within iteration-dependent periods, e.g., [11,13]. Specifically, the fact that a discontinuous function is $\mathscr{L}_{2\mathrm{e}}$-norm bounded does not show that the function is $\mathscr{L}_{\infty\mathrm{e}}$-norm bounded, which shows that the boundedness of estimates and inputs may not be guaranteed in the existing studies of conventional AILC. This work is dedicated to solving this problem, so the following objectives are given.

Problem formulation. Given an n-DOF robot manipulator operating within iteration-dependent periods and the desired trajectory described by $q_d(t) \in \mathbb{R}^n$, $\dot{q}_d(t) \in \mathbb{R}^n$, and $\ddot{q}_d(t) \in \mathbb{R}^n$, the objectives are as follows:

(a) The perfect tracking performance can be achieved in the presence of iteration-dependent periods and iteration-time-dependent uncertainties consisting of disturbances and unmodeled dynamics, i.e.,

$$\lim_{k\to\infty} T_k^{-1}\int_0^{T_k} |\tilde{q}_k^{(i)}(t)|\mathrm{d}\tau = \lim_{k\to\infty} T_k^{-1}\int_0^{T_k} |\dot{\tilde{q}}_k^{(i)}(t)|\mathrm{d}\tau = 0,\ \forall t \in [0, T_k], \tag{5}$$

where $\tilde{q}_k(t) \triangleq q_d(t) - q_k(t)$ and $\dot{\tilde{q}}_k(t) = \dot{q}_d(t) - \dot{q}_k(t)$.

(b) The uniform boundedness of system states, inputs, and estimates is guaranteed in the sense of the $\mathscr{L}_{\infty\mathrm{e}}$-norm, not just the $\mathscr{L}_{2\mathrm{e}}$-norm.

It is worth highlighting that when iteration-dependent periods occur, maybe only the $\mathscr{L}_{2\mathrm{e}}$-norm of estimates and inputs are achieved if the conventional AILC is applied, which implies that inputs may approach infinity at some time instant when the iteration tends to infinity. One of the main reasons why we cannot obtain the $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness of estimates and inputs in the framework of the conventional AILC is the non-continuity of estimates caused by iteration-dependent periods, for example, in [11].

Remark 1. According to the definition of the $\mathscr{L}_{p\mathrm{e}}$-norm in Section 1, the $\mathscr{L}_{2\mathrm{e}}$-norm boundedness can be obtained naturally from the $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness, but not vice versa.

To achieve the objectives (a) and (b), some assumptions are necessary.

Assumption 1. The identical initial condition is satisfied, i.e., $\|\tilde{q}_k(0)\| = \|\dot{\tilde{q}}_k(0)\| = 0,\ \forall k \in \mathbb{Z}_+$.

Remark 2. It is noted that Assumption 1 is presented in [17] and is necessary for the perfect tracking objective (5), because the initial states for all iterations do not respond to the controllers operating within $t \in [0, T_k]$. Moreover, if Assumption 1 is not satisfied, the rectification methods of initial states proposed in [13, 23] can be applied.

Assumption 2. There exists some $\sigma \in \mathbb{Z}_+$ such that

$$\{k : T_k = T_H\} \cap [k', k' + \sigma] \neq \emptyset, \quad \forall k' \in \mathbb{Z}_+. \tag{6}$$

Assumption 2 is presented in [27]. If Assumption 2 is satisfied, we have a persistent full-learning property: $0 \leqslant \psi(k) - k \leqslant \sigma$ for all $k \in \mathbb{Z}_+$, where $\psi(k) \triangleq \min\{k' \in \mathbb{Z}_+ : k' \geqslant k \text{ and } T_{k'} = T_H\}$. This means that the iteration interval between any two sequential iterations with a full-length period is uniformly bounded. As a special case, if $\sigma = 0$, then $T_k = T_H$, $\forall k \in \mathbb{Z}_+$, which indicates the iteration-independent full-length periods.

4 Main results of VMSs-based AILC

Before proceeding to the detailed design of AILC, the full-length period T_H is equally divided into $m \in \mathbb{Z}_+ \setminus \{0\}$ segments. Segment i shares the historical information stored in one VMS, which will help implement the segment-wise learning in subsequent controller design. The length of a unit segment can be defined by $T_U = \frac{T_H}{m}$, and segment i is denoted by $[(i-1)T_U, iT_U]$. With this definition, we can try to make each segment share the same virtual slot containing the historical information and relieve the memory storage pressure of the conventional point-wise AILC, which will be explained in subsequent Remark 3 in detail.

To achieve the objective mentioned in Section 3, the following adaptive learning controller is proposed for a rigid robot manipulator (2) that operates within $t \in [0, T_k]$:

$$u_k(t) = f(\ddot{q}_d(t), \dot{q}_k(t), \dot{q}_d(t))\hat{\beta}_k(t)\,\mathrm{sgn}(\dot{\tilde{q}}_k(t)) + k_c\tilde{q}_k(t) + k_d\dot{\tilde{q}}_k(t), \tag{7}$$

where k_c, $k_d \in \mathbb{R}$ are two positive parameters to be determined, $f(\ddot{q}_d(t), \dot{q}_k(t), \dot{q}_d(t))$ is defined as

$$f(\ddot{q}_d(t), \dot{q}_k(t), \dot{q}_d(t)) \triangleq \|\ddot{q}_d(t)\| + \|\dot{q}_k(t)\|\|\dot{q}_d(t)\| + 2, \tag{8}$$

$\hat{\beta}_k(t) \in \mathbb{R}$ is defined as

$$\hat{\beta}_k(t) = \begin{cases} \hat{\beta}_{k,1}(t), & \text{if } t \in [0, T_U], \\ \hat{\beta}_{k,i}(t), & \text{if } t \in ((i-1)T_U, iT_U],\ \forall i \in \mathbb{Z}_m \setminus \{1\}, \end{cases} \tag{9}$$

and $\hat{\beta}_{k,i}(t)$, $i \in \mathbb{Z}_m$, $t \in [(i-1)T_U, iT_U]$ denotes the estimate of β_i. Here, β_i represents the upper bound of uncertainties over segment i, i.e.,

$$\beta_i \triangleq \sup\nolimits_{t\in[(i-1)T_U, iT_U]} \max\{\beta_M(t), \beta_C(t), \beta_G(t), \beta_D(t)\}. \tag{10}$$

It is obtained that the relationship between i and t is

$$\frac{t}{T_U} \leqslant i \leqslant \frac{t}{T_U} + 1$$

according to (9) and (10). Specifically, we have two different estimates (i.e., $\hat{\beta}_{k,i}(t)$ and $\hat{\beta}_{k,i+1}(t)$) if $t = iT_U$ with $i \in \mathbb{Z}_m \setminus \{m\}$ because of the segment division for the segment-wise learning. This design makes the estimation process of two successive segments independent of each other. Moreover, note that $\hat{\beta}_{k,i}(t)$, $\forall i \in \mathbb{Z}_m$ are stored in m virtual memory slots, which are called VMSs in this work. Instead of requiring all historical information within $[0, T_H]$ for the conventional point-wise AILC, VMSs only need a set of m floating numbers.

For iteration k, the index of the segment where the operation ends is defined as $s_k \in \mathbb{Z}_m$ such that

$$(s_k - 1)T_U < T_k \leqslant s_k T_U. \tag{11}$$

It is obvious that s_k is iteration-dependent due to the change of T_k. The initial values stored in VMSs can be given by $\hat{\beta}_{0,i}(t) = 0$, $\forall i \in \mathbb{Z}_m$, $\forall t \in [0, T]$. The segment-wise learning law can be designed as, for $t \in [(i-1)T_U, iT_U]$ and $k \in \mathbb{Z}_+ \setminus \{0\}$,

$$\hat{\beta}_{k,i}(t) = \begin{cases} \hat{\beta}_{k-1,i}(iT_U), & \text{if } i > s_k \text{ and } t \in [(i-1)T_U, iT_U], \\ \hat{\beta}_{k,i}(T_k), & \text{if } i = s_k \text{ and } t \in (T_k, iT_U], \\ \hat{\beta}_{k-1,i}(iT_U) + \gamma \int_{(i-1)T_U}^{t} f(\ddot{q}_d(\tau), \dot{q}_k(\tau), \dot{q}_d(\tau))\|\dot{\tilde{q}}_k(\tau)\|_1 \mathrm{d}\tau, & \\ \qquad \text{if } i \leqslant s_k \text{ and } t \in [(i-1)T_U, \min\{iT_U, T_k\}], & \end{cases} \tag{12}$$

where $\gamma \in \mathbb{R}$ is a positive learning parameter. Based on (12), it is clear that VMSs-based learning operates on a segment-wise level rather than a point-wise level. Moreover, Figure 2 illustrates a diagram demonstrating the functionality of the proposed AILC equipped with VMSs.

Remark 3. According to (12), the number of the required historical data of the segment-wise learning strategy is m regardless of the control step size. Compared with conventional point-wise AILC requiring estimates for all $t \in [0, T_H]$, the VMSs-based segment-wise learning method requires less memory volume, which reduces the data storage pressure.

The following theorem shows that the proposed practical AILC equipped with VMSs can achieve two objectives mentioned in Section 3.

Theorem 1. For a rigid robot manipulator (2) under Assumptions 1 and 2, let the proposed AILC consisting of (7), (9), and (12) be applied. Then, the perfect tracking objective (5) is achieved. Moreover, not only the $\mathscr{L}_{2\mathrm{e}}$-norm boundedness, but also the $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness of all system signals can be guaranteed.

Proof. To establish this theorem, we adopt an energy function and define a candidate in the form of

$$E_{k,i}(t) \triangleq \gamma V_k(t) + \frac{1}{2}\tilde{\beta}_{k,i}^2(t),\ \forall t \in [(i-1)T_U, iT_U],\ i \in \mathbb{Z}_m, \tag{13}$$

where $\tilde{\beta}_{k,i}(t)$ is defined as

$$\tilde{\beta}_{k,i}(t) \triangleq \beta_i - \hat{\beta}_{k,i}(t), \tag{14}$$

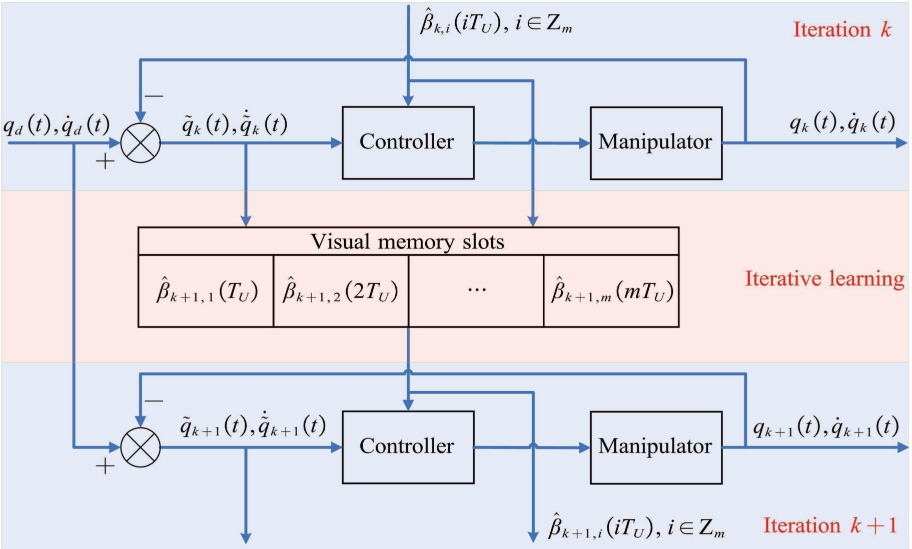

Figure 2 (Color online) Diagram of the proposed VMSs-based AILC.

and $V_k(t)$ is defined as

$$V_k(t) = \begin{cases} \frac{1}{2}\dot{\tilde{q}}_k^{\mathrm{T}}(t)M(q_k(t))\dot{\tilde{q}}_k + \frac{1}{2}k_c\tilde{q}_k^{\mathrm{T}}(t)\tilde{q}_k(t), & \text{if } t \leqslant T_k, \\ V_k(T_k), & \text{if } T_k < t \leqslant T_H. \end{cases}$$

Note that the learning stops at $t = T_k$, and the inclusion of the time interval $T_k < t \leqslant T_H$ in the analysis is merely for the sake of convenience, as it allows for a more comprehensive understanding of the system's behavior during a specific time frame.

Obviously, $V_k(t)$ is continuous along the time axis, but it may not be differentiable at T_k. To obtain the derivative of $V_k(t)$, we can separate the analysis into two cases: $t < T_k$ and $t > T_k$. Now, let us first investigate $\dot{V}_k(t)$, $\forall t < T_k$. It follows that

$$\dot{V}_k(t) = \dot{\tilde{q}}_k^{\mathrm{T}}(t)\left(M(q_k(t))\ddot{\tilde{q}}_k + \frac{1}{2}\dot{M}(q_k(t))\dot{\tilde{q}}_k\right) + k_c\dot{\tilde{q}}_k^{\mathrm{T}}(t)\tilde{q}_k(t), \tag{15}$$

where the symmetric property of $M(q_k(t))$ is considered. Eq. (15) can be further rewritten as

$$\begin{aligned}\dot{V}_k(t) = {} & \dot{\tilde{q}}_k^{\mathrm{T}}(t)M(q_k(t))\left(\ddot{q}_d(t) - \ddot{q}_k(t)\right) + \dot{\tilde{q}}_k^{\mathrm{T}}(t)\Big(\frac{1}{2}\dot{M}(q_k(t))\dot{\tilde{q}}_k(t) - C(q_k(t),\dot{q}_k(t))\dot{\tilde{q}}_k(t) \\ & + C(q_k(t),\dot{q}_k(t))\dot{\tilde{q}}_k(t) + k_c\tilde{q}_k(t)\Big),\end{aligned}$$

where $\ddot{\tilde{q}}_k(t) = \ddot{q}_d(t) - \ddot{q}_k(t)$ is applied. According to (2) and (4), we further have

$$\dot{V}_k(t) = \dot{\tilde{q}}_k^{\mathrm{T}}(t)\Big(M(q_k(t))\ddot{q}_d(t) + C(q_k(t),\dot{q}_k(t))\dot{q}_d(t) + G(q_k(t)) + k_c\tilde{q}_k(t) - u_k(t) - d_k(t)\Big).$$

Considering the definitions of $\beta_M(t)$, $\beta_C(t)$, $\beta_G(t)$, and $\beta_D(t)$, we have

$$\begin{aligned}\dot{V}_k(t) \leqslant {} & \|\dot{\tilde{q}}_k(t)\|_1\left(\beta_M(t)\|\ddot{q}_d(t)\| + \beta_C(t)\|\dot{q}_k(t)\|\|\dot{q}_d(t)\|\right) + \|\dot{\tilde{q}}_k(t)\|_1\left(\beta_G(t) + \beta_D(t)\right) \\ & + k_c\dot{\tilde{q}}_k^{\mathrm{T}}(t)\tilde{q}_k(t) - \dot{\tilde{q}}_k^{\mathrm{T}}(t)u_k(t),\end{aligned} \tag{16}$$

where $\|\dot{\tilde{q}}_k^{\mathrm{T}}(t)\| = \|\dot{\tilde{q}}_k(t)\| \leqslant \|\dot{\tilde{q}}_k(t)\|_1$ is applied. If Eqs. (8) and (10) are considered, then Eq. (16) can be written in the form of

$$\dot{V}_k(t) \leqslant \|\dot{\tilde{q}}_k(t)\|_1 f(\ddot{q}_d(t),\dot{q}_k(t),\dot{q}_d(t))\beta_i + k_c\dot{\tilde{q}}_k^{\mathrm{T}}(t)\tilde{q}_k(t) - \dot{\tilde{q}}_k^{\mathrm{T}}(t)u_k(t), \tag{17}$$

where $t \in ((i-1)T_U, \min\{iT_U, T_k\}]$ with $1 \leqslant i \leqslant s_k$.

Substituting (7) into (17), we have

$$\dot{V}_k(t) \leqslant \|\dot{\tilde{q}}_k(t)\|_1 f(\ddot{q}_d(t), \dot{q}_k(t), \dot{q}_d(t))\tilde{\beta}_{k,i}(t),\ \forall t \in ((i-1)T_U, \min\{iT_U, T_k\}],\ 1 \leqslant i \leqslant s_k. \tag{18}$$

Obviously, it is obtained that

$$\dot{V}_k(t) = 0,\ \forall t > T_k. \tag{19}$$

By resorting to the energy function candidate (13), we separate the subsequent analysis into the following four steps.

Step 1. Study (13) **at** $t = iT_U$**.** For $t = iT_U$, we have

$$E_{k,i}(iT_U) = \gamma V_k(iT_U) + \frac{1}{2}\tilde{\beta}_{k,i}^2(iT_U). \tag{20}$$

It is obtained that the difference between two consecutive iterations is defined as, for all $i \in \mathbb{Z}_m$,

$$\begin{aligned}\Delta E_{k,i}(iT_U) &= E_{k,i}(iT_U) - E_{k-1,i}(iT_U)\\ &= E_{k,i}((i-1)T_U) + \int_{(i-1)T_U}^{iT_U} \dot{E}_{k,i}(t)\mathrm{d}t - E_{k-1,i}(iT_U)\end{aligned} \tag{21}$$

according to (20). Note that $\dot{E}_{k,i}(t)$ may not exist when $(i-1)T_U < T_k < iT_U$. Fortunately, this fact does not affect the integral because $E_{k,i}(t)$ is continuous within $[(i-1)T_U, iT_U]$ according to (13).

From (12), it is derived that

$$\frac{1}{2}\tilde{\beta}_{k-1,i}^2(iT_U) = \frac{1}{2}\tilde{\beta}_{k,i}^2((i-1)T_U), \tag{22}$$

which implies

$$E_{k,i}((i-1)T_U) - E_{k-1,i}(iT_U) = \gamma V_k((i-1)T_U) - \gamma V_{k-1}(iT_U). \tag{23}$$

Substituting (23) into (21), we have

$$\Delta E_{k,i}(iT_U) = \int_{(i-1)T_U}^{iT_U} \left(\tilde{\beta}_{k,i}(t)\dot{\tilde{\beta}}_{k,i}(t) + \gamma \dot{V}_k(t)\right)\mathrm{d}t + \gamma V_k((i-1)T_U) - \gamma V_{k-1}(iT_U) \tag{24}$$

based on (13). Considering (12) again, we obtain

$$\tilde{\beta}_{k,i}(t)\dot{\tilde{\beta}}_{k,i}(t) = -\gamma\tilde{\beta}_{k,i}(t) f(\ddot{q}_d(t), \dot{q}_k(t), \dot{q}_d(t))\|\dot{\tilde{q}}_k(t)\|_1, \tag{25}$$

if $i \leqslant s_k$ and $t \in [(i-1)T_U, \min\{iT_U, T_k\}]$, otherwise $\tilde{\beta}_{k,i}(t)\dot{\tilde{\beta}}_{k,i}(t) = 0$. This fact, together with (18), (19), and (24), leads to

$$\Delta E_{k,i}(iT_U) \leqslant \gamma V_k((i-1)T_U) - \gamma V_{k-1}(iT_U). \tag{26}$$

Additionally, according to the above analysis, it can be verified easily that $E_{0,i}(t)$ is $\mathscr{L}_{\infty\mathrm{e}}$-norm bounded for any $i \in \mathbb{Z}_m$ and $t \in [(i-1)T_U, iT_U]$, which is omitted here.

Step 2. The convergence of $V_k(iT_U)$ **with** $i = 1$**.** According to Assumption 1, $V_k(0) = 0$ holds for all $k \in \mathbb{Z}_+$. Therefore, Eq. (26) can be rewritten as

$$\Delta E_{k,1}(T_U) \leqslant -\gamma V_{k-1}(T_U) \leqslant 0, \tag{27}$$

which implies that $E_{k,1}(T_U)$ is non-increasing along the iteration axis and thus $E_{k,1}(T_U) < \infty$. Considering the positiveness of $E_{k,i}(t)$ defined in (13), we consequently have $\lim_{k\to\infty} V_{k-1}(T_U) = 0$ and $\sum_{k=1}^{\infty} V_{k-1}(T_U) < \infty$ according to (27).

Step 3. The convergence of $V_k(iT_U)$ **with** $i > 1$**.** Now, let us try to accumulate (26) from $k = 1$ to $k = \infty$. Although $E_{k,i}(iT_U)$ may not be non-increasing along the iteration axis, it is obtained that, for any $i \in \mathbb{Z}_m \setminus \{1\}$, $\lim_{k\to\infty} V_{k-1}(iT_U) = 0$ holds if $\sum_{k=1}^{\infty} V_k((i-1)T_U) < \infty$. This fact can be easily verified by using the famous mathematical induction with the results of Step 2. The analysis can be omitted due to the similarity with Step 2. As a result, we have

$$\lim_{k\to\infty} V_{k-1}(iT_U) = 0,\ \forall i \in \mathbb{Z}_m \cup \{0\} \tag{28}$$

and

$$E_{k,i}(iT_U) < \infty,\ \forall i \in \mathbb{Z}_m \cup \{0\}, \tag{29}$$

where Assumption 1 is considered.

Step 4. Study (13) **for** $t \in ((i-1)T_U, iT_U),\ i \in \mathbb{Z}_m$ **and** $i \in \mathbb{Z}_m$**.** Based on the results of Steps 2 and 3, we know that perfect tracking is achieved for $t = iT_U$ and all $i \in \mathbb{Z}_m \cup \{0\}$ when k tends to the infinity. Now, let us check the boundedness of $E_{k,i}(t)$ and the tracking performance within $((i-1)T_U, iT_U)$ with $i \in \mathbb{Z}_m$.

According to (23) and (28), it is obtained that

$$\lim_{k\to\infty} (E_{k-1,i}(iT_U) - E_{k,i}((i-1)T_U)) = 0,\ \forall i \in \mathbb{Z}_m. \tag{30}$$

In addition, based on (21) and (26), we have

$$\lim_{k\to\infty} (E_{k,i}(iT_U) - E_{k-1,i}(iT_U)) = 0,\ \forall i \in \mathbb{Z}_m. \tag{31}$$

Adding (30) and (31), we obtain

$$\lim_{k\to\infty} (E_{k,i}(iT_U) - E_{k,i}((i-1)T_U)) = 0,\ \forall i \in \mathbb{Z}_m, \tag{32}$$

which, together with (20) and (28), leads to

$$\lim_{k\to\infty} \left(\tilde{\beta}_{k,i}(iT_U) - \tilde{\beta}_{k,i}((i-1)T_U)\right) = 0 \tag{33}$$

or/and

$$\lim_{k\to\infty} \left(\tilde{\beta}_{k,i}(iT_U) + \tilde{\beta}_{k,i}((i-1)T_U)\right) = 0. \tag{34}$$

Actually, Eq. (33) holds, but Eq. (34) does not always hold. This can be proven by using the reduction to absurdity. Specifically, on the one hand, suppose that only Eq. (34) holds, then

$$\begin{aligned}&\lim\nolimits_{k\to\infty}(\hat{\beta}_{k,i}(iT_U) + \hat{\beta}_{k,i}((i-1)T_U)) = 2\beta_i,\\ &\lim\nolimits_{k\to\infty}(\hat{\beta}_{k+1,i}(iT_U) + \hat{\beta}_{k+1,i}((i-1)T_U)) = 2\beta_i.\end{aligned} \tag{35}$$

Additionally, according to (12), we have $\hat{\beta}_{k,i}((i-1)T_U) \leqslant \hat{\beta}_{k,i}(iT_U) = \hat{\beta}_{k+1,i}((i-1)T_U) \leqslant \hat{\beta}_{k+1,i}(iT_U)$. This leads to a contradiction with (35) unless $\tilde{\beta}_{k,i}(iT_U) = \tilde{\beta}_{k,i}((i-1)T_U) = 0$. On the other hand, suppose that both Eqs. (33) and (34) hold, the same contradiction occurs again. Therefore, Eq. (33) always holds.

Considering the continuity of $\dot{\tilde{q}}_k(t)$, we can obtain

$$\lim_{k\to\infty} \|\dot{\tilde{q}}_k(t)\| = 0,\ \forall t \in [0, T_k] \tag{36}$$

based on (12), (28), and (33). Further, Eqs. (12) and (36) lead to, for all $i \in \mathbb{Z}_m$,

$$\lim_{k\to\infty} \left(\tilde{\beta}_{k,i}(t) - \tilde{\beta}_{k,i}((i-1)T_U)\right) = 0,\ \forall t \in [(i-1)T_U, T_U],$$

which, together with (13), (18), and (25), implies

$$\lim_{k\to\infty} \|\tilde{q}_k(t)\| = 0,\ \forall t \in [0, T_k]. \tag{37}$$

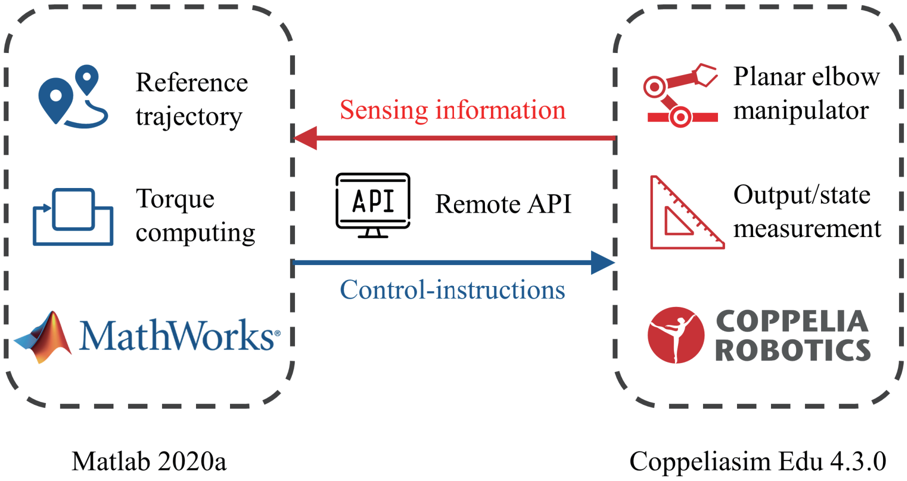

Figure 3 (Color online) Illustration of a visual simulation platform consisting of Matlab 2020a and Coppeliasim Edu 4.3.0.

According to (36) and (37), it is not hard to verify that Eq. (5) is achieved.

According to Assumption 2, when k tends to infinity, the number of $T_k = T_H$ also tends to infinity. From (28), (36), and (37), the perfect attitude tracking objective (5) is achieved. Moreover, a natural result from (29) is $V_k(iT_U) < \infty$ and $\tilde{\beta}^2_{k,i}(iT_U) < \infty$ hold for all $i \in \mathbb{Z}_m \cup \{0\}$ and $k \in \mathbb{Z}_+$, which implies $\tilde{\beta}^2_{k+1,i}((i-1)T_U) < \infty,\ \forall i \in \mathbb{Z}_m,\ \forall k \in \mathbb{Z}_+ \cup \{-1\}$ (hold naturally for $k = -1$) according to (12) and thus

$$E_{k,i}((i-1)T_U) < \infty,\ \forall i \in \mathbb{Z}_m,\ \forall k \in \mathbb{Z}_+. \tag{38}$$

Based on the continuity of $E_{k,i}(t)$ within $[(i-1)T_U, iT_U]$, Eqs. (29) and (38) lead to $E_{k,i}(t) < \infty, \forall i \in \mathbb{Z}_m \cup \{0\},\ \forall t \in [(i-1)T_U, iT_U]$, which implies

$$\begin{aligned} &V_k(t) < \infty,\ \forall t \in [0, T_H],\ \forall k \in \mathbb{Z}_+, \\ &\hat{\beta}_{k,i}(t) < \infty,\ \forall i \in \mathbb{Z}_m,\ \forall k \in \mathbb{Z}_+,\ \forall t \in [(i-1)T_U, iT_U]. \end{aligned}$$

Therefore, all signals, including states, estimates, and inputs, are $\mathscr{L}_{\infty\mathrm{e}}$-norm bounded. The $\mathscr{L}_{2\mathrm{e}}$-norm boundedness of them can be naturally guaranteed according to the relationship between these two norms in (1). The proof of Theorem 1 is complete.

Remark 4. As stated in [14], although the $\mathscr{L}_{2\mathrm{e}}$-norm boundedness can be achieved in the literature of the conventional AILC, the $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness is also necessary for the actual applications. Although some mechanisms based on the prior information of uncertainties are designed for restricting the estimates and/or inputs, e.g., the projection mechanisms, actually they are implicitly contrary to the assumption of uncertainties with known bounds [13]. In comparison, the proposed method can guarantee the $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness of systems without requiring extra prior information on uncertainties.

Remark 5. This work considers that the states of robot manipulators can be accessed directly. This is reasonable in many cases, as many sensors can be used to measure the states of robot manipulators [18]. However, there are some cases where the states of robot manipulators cannot be accessed directly. For example, if a robot is operating in a harsh environment, the sensors may be damaged. In these cases, it is necessary to develop other approaches to estimate the states of the robot.

5 A solution to the visual simulation

A visual manipulator simulation platform consisting of two trustworthy software (i.e., Coppeliasim Edu 4.3.0 and Matlab 2020a) is constructed, and the communication between both software is completed with the ZeroMQ remote application program interface (API), which can let Matlab interact with CoppeliaSim in a stepped way (i.e., synchronized with each simulation step). The platform is illustrated in Figure 3. Specifically, Coppeliasim provides an integrated development environment including open dynamics engine (ODE) and modeled proximity sensors, which offer us the trusted planar elbow manipulator equipped with angle/angular velocity sensors and the gravitational field. In Matlab, we can utilize

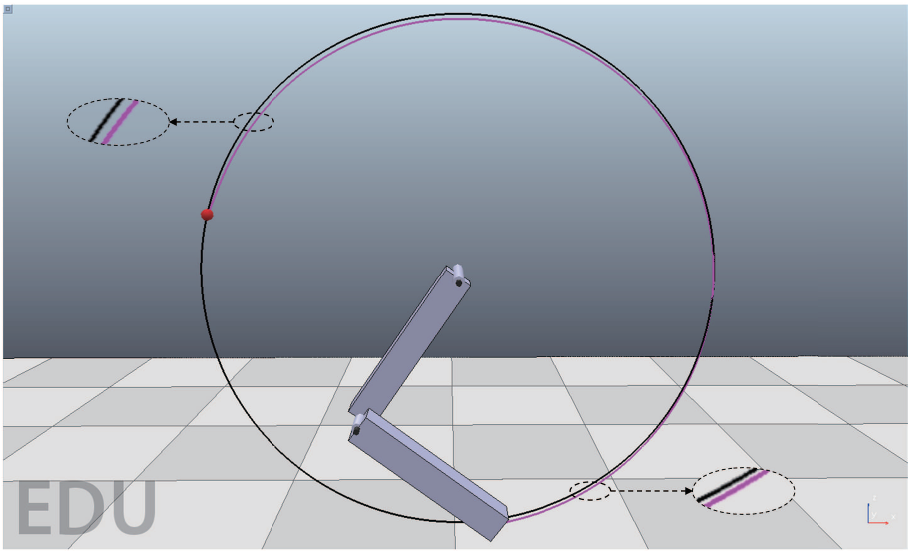

Figure 4 (Color online) Tracking performance for $k = 0$ shown in Coppeliasim. Red dot: the starting point of the actual trajectory. Purple line: the actual trajectory of link 2's end. Black circle: the reference trajectory of link 2's end.

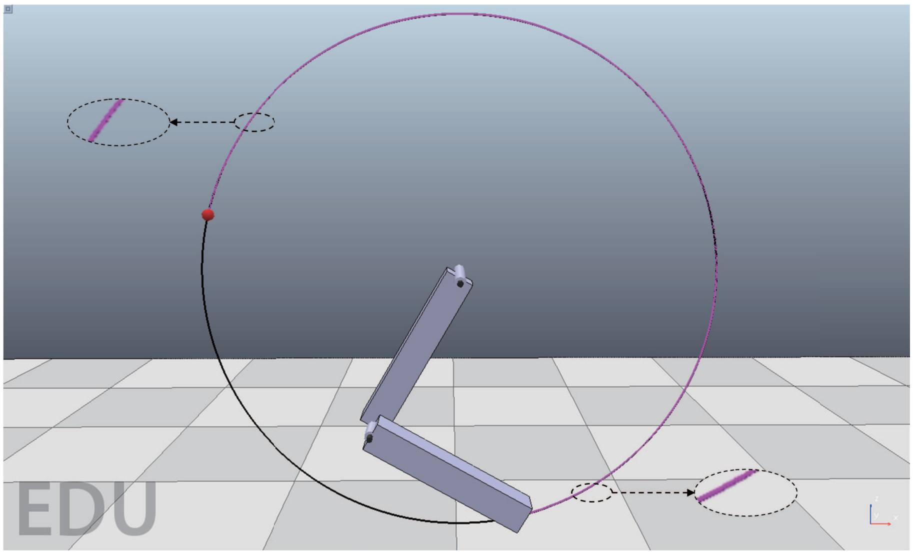

Figure 5 (Color online) Tracking performance for $k = 8$ shown in Coppeliasim. Red dot: the starting point of the actual trajectory. Purple line: the actual trajectory of link 2's end. Black circle: the reference trajectory of link 2's end.

the reference trajectory and the output/state measurements from Coppeliasim to calculate the torque commands, and transfer them to Coppeliasim to drive the joints in high-level control (HLC).

Now we consider the two-DOF robot manipulator illustrated in Figure 1. The values of the parameters are listed as follows. $m_1 = m_2 = 1\,\mathrm{kg}$, $l_1 = l_2 = 0.5\,\mathrm{m}$, $l_{c1} = l_{c2} = 0.25\,\mathrm{m}$, $I_1 = I_2 = 0.1\,\mathrm{kg}\cdot\mathrm{m}^2$, $g = 9.81\,\mathrm{m/s}^2$. The external disturbances are set as $d_1 = d_2 = \mathrm{rand}(k)\sin(t)$, where $0 \leqslant \mathrm{rand}(k) \leqslant 1$ can be random and iteration-dependent. The desired trajectory is described by $q_d(t) = [\frac{2}{3}\pi\cos(\frac{1}{2}\pi t), \frac{\pi}{2}]^{\mathrm{T}}$, $\dot{q}_d(t)$ and $\ddot{q}_d(t)$ can be derived from the evolution of $q_d(t)$ along the time axis. For iteration-dependent periods, we assign $T_H = 2\,\mathrm{s}$, $T_L = 1.5\,\mathrm{s}$, and $\sigma = 1$, which means that $T_k = T_H$ and $T_k = T_L$ are alternate. The communication step is set as $0.0001\,\mathrm{s}$.

The controller parameters in (7) are $k_c = 60$ and $k_d = 20$, the learning parameter in (12) is $\gamma = 10$, and the number of slots is 20. It is important to note that the values of k_c, k_d, and γ are closely related to the sampling time. They can be roughly estimated using the method described in [28]. The corresponding simulation results are illustrated in Figures 4–9. The visual results for $k = 0$ and $k = 8$ are illustrated in Figures 4 and 5, respectively, which intuitively show that the tracking of the reference trajectory of a robot manipulator for $k = 8$ is better than that for $k = 0$. Let us further investigate the

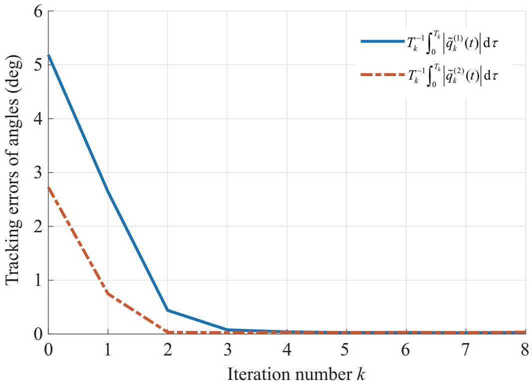

Figure 6 (Color online) Tracking errors of angles versus the number of iterations for links 1 and 2.

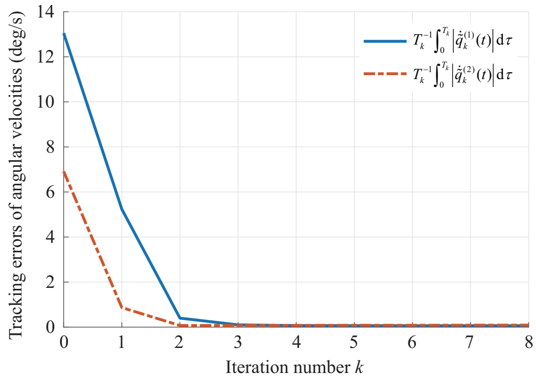

Figure 7 (Color online) Tracking errors of angular velocities versus the number of iterations for links 1 and 2.

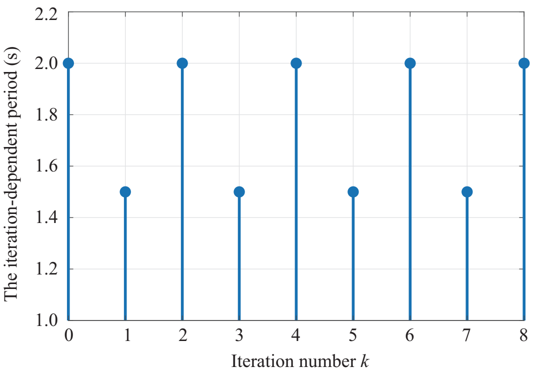

Figure 8 (Color online) Illustration of iteration-dependent periods.

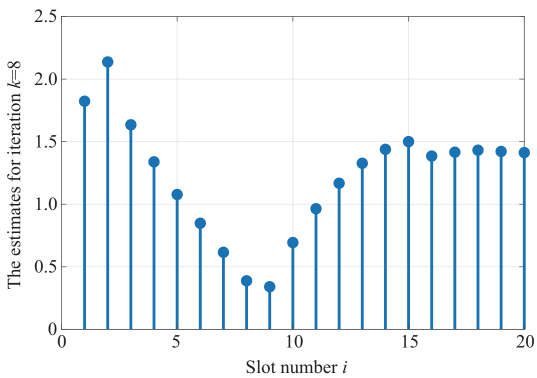

Figure 9 (Color online) Estimates for iteration $k = 8$ stored in VMSs.

tracking performance from $k = 0$ to $k = 8$. The tracking performance of angles and angular velocities are illustrated in Figures 6 and 7, respectively. It is obvious that the tracking performance can be refined gradually via the proposed AILC method from iteration to iteration. All tracking errors are small enough when $k = 8$, which is consistent with the statement in Theorem 1. This shows the good performance of the proposed AILC. Additionally, the results show that the proposed AILC works well when iteration-dependent periods occur, which can be found in Figure 8. Additionally, the boundedness of estimates can be observed in Figure 9, which cannot be obtained in the framework of conventional point-wise AILC.

Finally, one thing to be reminded of is that the definition of joint angles in the default settings of Coppeliasim is different from that in the classical Euler-Lagrange equation of the current work. Therefore, in the construction of the visual platform consisting of Coppeliasim and Matlab, both definitions in two software should be standardized.

6 Conclusion

In this paper, a novel and practical AILC equipped with VMSs has been developed for robot manipulators. To address the $\mathscr{L}_{\infty\mathrm{e}}$-norm boundedness problem encountered in point-wise AILC, the segment-wise learning method has been developed with the help of VMSs. Except for achieving the perfect tracking objective, this method is able to simultaneously guarantee the $\mathscr{L}_{\infty\mathrm{e}}$-norm and $\mathscr{L}_{2\mathrm{e}}$-norm boundedness of all signals of the plant. Compared with conventional point-wise AILC, the proposed method requires less memory storage for historical tracking information. Moreover, it has been demonstrated that the proposed method is applicable to robot manipulators regardless of iteration-dependent periods. The effectiveness of the proposed AILC has been verified through a visual simulation platform conducted on a platform combining Coppeliasim and Matlab, showcasing its exceptional tracking and learning capabilities. Furthermore, two intriguing research directions are the determination of an appropriate segment number and the complete elimination of the impact of the previous segment's tracking on the subsequent segment.

Acknowledgements This work was supported in part by National Natural Science Foundation of China (Grant Nos. U2333215, 62273018), National Key Research and Development Program of China (Grant No. 2021YFB2601703), and Science and Technology on Space Intelligent Control Laboratory (Grant No. HTKJ2022KL502006).

References

1 Bu X, Hou Z. Adaptive iterative learning control for linear systems with binary-valued observations. IEEE Trans Neural Netw Learn Syst, 2018, 29: 232–237
2 Hui Y, Chi R, Huang B, et al. Data-driven adaptive iterative learning bipartite consensus for heterogeneous nonlinear cooperation-antagonism networks. IEEE Trans Neural Netw Learn Syst, 2023, 34: 8262–8270
3 Nekoo S R, Acosta J Á, Heredia G, et al. A PD-type state-dependent Riccati equation with iterative learning augmentation for mechanical systems. IEEE CAA J Autom Sin, 2022, 9: 1499–1511
4 Wang M, Zhang Y L, Dong H F, et al. Trajectory tracking control of a bionic robotic fish based on iterative learning. Sci China Inf Sci, 2020, 63: 170202
5 Liang X Q, Zhao H, Li X F, et al. Force tracking impedance control with unknown environment via an iterative learning algorithm. Sci China Inf Sci, 2019, 62: 050215
6 Stolfi A, Gasbarri P, Sabatini M. A combined impedance-PD approach for controlling a dual-arm space manipulator in the capture of a non-cooperative target. Acta Astronaut, 2017, 139: 243–253
7 Han H, Wu X, Qiao J. Design of robust sliding mode control with adaptive reaching law. IEEE Trans Syst Man Cybern Syst, 2020, 50: 4415–4424
8 Fu Q, Gu P P, Wu J R. Iterative learning control for one-dimensional fourth order distributed parameter systems. Sci China Inf Sci, 2017, 60: 012204
9 Bristow D A, Tharayil M, Alleyne A G. A survey of iterative learning control. IEEE Control Syst Mag, 2006, 26: 96–114
10 Meng D, Zhang J. Robust tracking of nonrepetitive learning control systems with iteration-dependent references. IEEE Trans Syst Man Cybern Syst, 2021, 51: 842–852
11 Shen D, Xu J X. Adaptive learning control for nonlinear systems with randomly varying iteration lengths. IEEE Trans Neural Netw Learn Syst, 2019, 30: 1119–1132
12 Shen D, Li X. A survey on iterative learning control with randomly varying trial lengths: Model, synthesis, and convergence analysis. Annu Rev Control, 2019, 48: 89–102
13 Chen Q, Chen K, Shi H, et al. Error tracking iterative learning control for robot manipulators with varying iterative lengths (in Chinese). Acta Automat Sin, 2023, 49: 2594–2604
14 Dixon W E, Chen J. Comments on "A composite energy function-based learning control approach for nonlinear systems with time-varying parametric uncertainties". IEEE Trans Automat Contr, 2003, 48: 1671–1672
15 Xu J-X, Tan Y. A composite energy function-based learning control approach for nonlinear systems with time-varying parametric uncertainties. IEEE Trans Automat Contr, 2002, 47: 1940–1945
16 Xu J-X, Tan Y. Authors' reply [to comments on "A composite energy function-based learning control approach for nonlinear systems with time-varying parametric uncertainties"]. IEEE Trans Automat Contr, 2003, 48: 1672–1674
17 Li X, Liu C, Chen B, et al. Robust adaptive learning-based path tracking control of autonomous vehicles under uncertain driving environments. IEEE Trans Intell Transp Syst, 2022, 23: 20798–20809
18 Tayebi A. Adaptive iterative learning control for robot manipulators. Automatica, 2004, 40: 1195–1203
19 Zhang F, Meng D, Li X. Robust adaptive learning for attitude control of rigid bodies with initial alignment errors. Automatica, 2022, 137: 110024
20 Zhang F, Meng D, Li X. Chattering-free adaptive iterative learning for attitude tracking control of uncertain spacecraft. Automatica, 2023, 151: 110902
21 Chien C J, Tayebi A. Further results on adaptive iterative learning control of robot manipulators. Automatica, 2008, 44: 830–837
22 French M, Rogers E. Non-linear iterative learning by an adaptive Lyapunov technique. Int J Control, 2000, 73: 840–850
23 Jin X. Iterative learning control for non-repetitive trajectory tracking of robot manipulators with joint position constraints and actuator faults. Intl J Adaptive Control Signal, 2017, 31: 859–875
24 Sun T R, Cheng L, Peng L, et al. Learning impedance control of robots with enhanced transient and steady-state control performances. Sci China Inf Sci, 2020, 63: 192205
25 Wang H. Adaptive control of robot manipulators with uncertain kinematics and dynamics. IEEE Trans Automat Contr, 2017, 62: 948–954
26 Spong M W, Hutchinson S, Vidyasagar M. Robot Modeling and Control. 2nd ed. Hoboken: John Wiley & Sons, 2020
27 Meng D, Zhang J. Deterministic convergence for learning control systems over iteration-dependent tracking intervals. IEEE Trans Neural Networks Learn Syst, 2018, 29: 3885–3892
28 Wie B, Weiss H, Arapostathis A. Quarternion feedback regulator for spacecraft eigenaxis rotations. J Guidance Control Dyn, 1989, 12: 375–380