mathematics

Article

A Novel Predefined Time PD-Type ILC Paradigm for Nonlinear Systems

Chun-Wu Yin [1], Saleem Riaz [2], Haider Zaman [3], Nasim Ullah [4,*], Vojtech Blazek [5,*], Lukas Prokop [5] and Stanislav Misak [5]

1 School of Information and Control Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China
2 School of Automation, Northwestern Polytechnical University, Xi'an 710129, China
3 Electronics Engineering Department, University of Engineering and Technology Peshawar, Peshawar 25000, Pakistan
4 Department of Electrical Engineering, College of Engineering, Taif University, Taif 11099, Saudi Arabia
5 ENET Centre, VSB—Technical University of Ostrava, 708 00 Ostrava, Czech Republic
* Correspondence: nasimullah@tu.edu.sa (N.U.); vojtech.blazek@vsb.cz (V.B.)

Abstract: Intelligent robotics has drawn a great deal of attention due to its high precision, stability, and reliability, which are the basic key factors for industrial automation. This paper proposes an iterative learning control (ILC) technique with predefined-time convergence as a solution to an applied engineering problem, namely, that local time cannot be preset when a second-order nonlinear system undertakes control of the accurate tracking of local time under any initial iterative value. A time-varying sliding surface with an initial value of zero was designed, and it was theoretically proven that the trajectory tracking error in the sliding surface could converge to zero within a predefined time. The iterative control problem of trajectory tracking was thus changed to an iterative control problem of time-varying sliding-mode surface tracing with a starting value of zero. A PD-type closed-loop ILC with a time-varying sliding mode surface was designed such that the trajectory tracking error converged and stabilized on the sliding mode surface after a finite number of learning iterations. The control goal for the system's output was the ability to track the desired trajectory accurately within a predefined time interval, and it was achieved by combining this with the predefined time convergence characteristics of the time-varying sliding mode surface. Numerical simulation of trajectory tracking control of a repetitive motion manipulator was used to verify the effectiveness of the proposed controller and its robustness in the face of external disturbances.

Keywords: iterative learning control; sliding mode control; predefined-time convergence; time-varying sliding mode surface; robotic arm

MSC: 393D05; 37N35

Citation: Yin, C.-W.; Riaz, S.; Zaman, H.; Ullah, N.; Blazek, V.; Prokop, L.; Misak, S. A Novel Predefined Time PD-Type ILC Paradigm for Nonlinear Systems. *Mathematics* **2023**, *11*, 56. https://doi.org/10.3390/math11010056

Academic Editor: António Lopes

Received: 7 November 2022
Revised: 16 December 2022
Accepted: 18 December 2022
Published: 23 December 2022

1. Introduction

In engineering applications such as industrial recurrent production, hard disk drive control, and construction robot wall building, the output of the repetitive motion control system is required to move a mechanical arm strictly according to the desired trajectory within a finite time interval $[0, T]$. When the initial value of trajectory tracking error is zero, iterative learning control (ILC) is implemented. In short, ILC can ensure that the system output fully tracks the desired trajectory [1,2], but in practical engineering applications it is challenging to strictly locate the initial state of the controlled system at the initial position of the desired trajectory. The arbitrary initial value of iterative learning control can only ensure that the system output accurately tracks the desired trajectory in local time $[T_s, T]$, but the existing ILC control strategy cannot be preset or estimated in the bound of time T_s.

This restricts the application of ILC in practical engineering. Therefore, there is a strong need to design an iterative learning control algorithm with a predefined time of T_s.

Traditional ILC control theory is predicated on the assumption that the controlled system's iterative starting value deviation is zero. When the controlled system is satisfied with the iterative initial value constraint conditions, the output of the controlled system can be in a given time interval, in strict accordance with the desired trajectory [3–5] (which is perfect tracking), but in practical engineering applications, it is difficult to meet every time constraint with a zero initial iteration value deviation [6]. Thus, the engineering application of ILC theory is limited. Scholars have confirmed, through theoretical analysis and experimental verification, that when the deviation between the initial value of the controlled system and the desired trajectory is a fixed value, or when the initial value of the system and the desired trajectory satisfy a certain law, this value of the system can converge to the initial value of the desired trajectory, for which the ILC algorithm can also ensure that the output of the controlled system follows the desired trajectory [7,8] (for instance, the tracking accuracy in fractional order control [9,10] and optimal learning control [11]). This result relaxes the strict requirement that the iterative initial value variation of the conventional ILC must be zero [12,13], but it is still compatible with real-world engineering applications. When a difference arises, only iterations of the ILC control algorithm that satisfy the initial value criteria will be able to fulfil the needs of the practical engineering application.

Scholars have proposed control strategies such as model predictive control, the initial value correction method, the boundary layer method, and the attractor method [14–16], but the initial value correction method involves determining the delay factor in advance [17,18]. The boundary layer in the boundary layer method is asymptotically convergent, which means that the trajectory tracking error in the boundary layer can only converge to zero when time tends to infinity, resulting in low trajectory tracking accuracy [19]. The attractor design in the attractor method has certain limitations, and some attractor control strategies involve redesigning the desired trajectory [20].When the initial value of the tracking error between the system output and the desired trajectory is $\boldsymbol{e}_k(0) \equiv 0$, after a finite number of iterations the system output has the full ability to follow the desired trajectory within the finite time interval, $[0, T]$. However, when the initial value of the tracking error between the system output and the desired trajectory is $\boldsymbol{e}_k(0) \neq 0$, it means that particular system is able only to track the desired signal within the local time interval $[T_s, T]$, which is $\boldsymbol{e}_k(t) \equiv 0, t \in [T_s, T]$. Although the existing control strategies to suppress the initial value of any iteration can solve the iterative learning convergence problem under the initial value of any iteration, they cannot estimate or even set the time, T_s, to achieve local convergence in advance. In some practical engineering applications, it is required that the system output accurately tracks the desired trajectory before the given time, T_s. For example, when a construction robot performs construction processes such as concrete troweling or wall laying, the mechanical arm must reach the desired trajectory before the given time, T_s, and repeat the movement strictly according to the desired trajectory to ensure the smoothness of concrete troweling or the uniformity of wall tiles. Failure to achieve this can lead to major economic losses for the construction industry, as well as raise the risk of building collapse. Despite the importance of determining and presetting the local convergence time in many engineering applications, very little work has been done on the iterative learning control theory in regard to predefined-time convergence. At the same time, while many current iterative learning control strategies under arbitrary iterative initial values have been projected mainly for first-order systems, there are relatively few publications on iterative learning control strategies for second-order nonlinear systems under arbitrary iterative initial values.

This paper will focus on second-order nonlinear systems with repetitive motion, and propose a PD-type closed-loop iterative learning control strategy based on the predefined-time convergence sliding mode surface, aiming to show that the controlled system under any initial value can not only follow a local trajectory for accurate tracking, but also predetermine the local convergence time, T_s, in advance. The main innovations and contribution of this study can be summarized as follows:

1. Provides Lyapunov stability criterion for the stability of nonlinear systems within a predefined time and describes the theoretical proof under the given conditions.
2. Presents a design for a time-varying sliding mode surface with predefined time convergence characteristics in which the convergence time of the trajectory tracking error located in the sliding mode surface can be preset, bringing the advantage that the convergence time is not affected by the controlling constraints or the initial value of the iteration.
3. Converts the trajectory tracking control problem, where the initial value of the trajectory tracking error is not zero, into a sliding mode surface tracking control problem in which the initial value of the sliding mode surface being zero. Establishes a bridge between the iterative learning control theory with an arbitrary iterative initial value and the same iterative initial value.
4. The iterative learning control strategy not only solves the problem of arbitrary iterative initial value suppression and simplifies the theoretical proof of the convergence of iterative learning, it also achieves the engineering application of the system output, accurately tracking the desired trajectory within a preset local time.

The remainder of this paper is as follows. Section 2 presents the control problem formulation and also describes several lemmas for iterative learning convergence proof. Section 3 proposes an arbitrary initial value suppression strategy based on the predefined time convergence sliding mode control principle, mentioning its principles. The Lyapunov stability criterion for predefined time convergence of nonlinear systems is given, and a design for a sliding mode surface with the character of predefined time convergence and initial value of zero is presented. The main results of iterative convergence are discussed in Section 4, which also demonstrates the predefined time convergence condition for a PD-type ILC. In Section 5, the effectiveness of the proposed nonlinear control strategy is illustrated by simulations for a robotic system, the results of which are briefly explained. Finally, Section 6 presents the conclusions.

2. Control Problem Descriptions

Consider the following second-order nonlinear system with repetitive motion characteristics:

$$\begin{cases} \dot{\boldsymbol{x}}_{1k}(t) = \boldsymbol{x}_{2k}(t) \\ \dot{\boldsymbol{x}}_{2k}(t) = \boldsymbol{f}(\boldsymbol{x}_k(t), t) + \boldsymbol{B}(t)\boldsymbol{u}_k(t) \\ \boldsymbol{y}_k(t) = \boldsymbol{x}_{1k}(t) \end{cases} \tag{1}$$

where $\boldsymbol{x}_k(t) = [\boldsymbol{x}_{1k}(t), \boldsymbol{x}_{2k}(t)]^T$ indicates the state variable, $\boldsymbol{y}_k(t) \in \boldsymbol{R}^m$ is the output variable, $\boldsymbol{u}_k(t) \in \boldsymbol{R}^l$ is the control input variable, k represents the number of iterations, and $t \in [0, T]$, $\boldsymbol{B}(t)$ is the bounded function matrix of appropriate dimension. The function $\boldsymbol{f}(\boldsymbol{x}_k(t), t)$ satisfies the Lipschitz condition with respect to the state variable, $\boldsymbol{x}_k(t)$, in the time interval $t \in [0, T]$. That is means there is a constant,$M_1 > 0$, and function $\boldsymbol{f}(\boldsymbol{x}_k(t), t)$ satisfies

$$||\boldsymbol{f}(\boldsymbol{x}_k(t), t) - \boldsymbol{f}(\boldsymbol{x}_{\mathrm{d}}(t), t)|| \leq M_1||\boldsymbol{x}_k(t) - \boldsymbol{x}_{\mathrm{d}}(t)|| \tag{2}$$

The control objective: Let the desired trajectory of the second-order nonlinear system (1) be $\boldsymbol{y}_{\mathrm{d}}(t)$ in an application environment where the iterative initial value, $\boldsymbol{y}_k(0)$, of the second-order nonlinear system (1) cannot be strictly located at the initial value, $\boldsymbol{y}_{\mathrm{d}}(0)$, of the desired trajectory. Design an iterative learning controller, $\boldsymbol{u}_k(t)$, to make the output of system (1) precisely track the desired trajectory, $\boldsymbol{y}_k(t)$, over a predefined-time interval, $[T_s, T](0 < T_s < T)$.

The tracking error between the system output, $\boldsymbol{y}_k(t)$, and the target trajectory, $\boldsymbol{y}_\mathrm{d}(t)$, may be determined as follows:

$$\boldsymbol{e}_k(t) = \boldsymbol{y}_k(t) - \boldsymbol{y}_\mathrm{d}(t) \tag{3}$$

Below are some lemmas for iterative learning convergence proof:

Lemma 1 . *Let $w(t), b(t), a(t)$ be a continuous function defined on the interval $[0, T]$, and $a(t) > 0$. If* [21]

$$w(t) \le b(t) + \int_0^t a(\tau) w(\tau) \mathrm{d}\tau \tag{4}$$

then $w(t) \le b(t) + \int_0^t a(\tau) b(\tau) e^{\int_\tau^t a(\lambda) d\lambda} d\tau$.

Lemma 2 . *Suppose that the mentioned function, $\boldsymbol{O}(\boldsymbol{\xi})(t)$, for the time interval $t \in [\Delta, T]$ mollifies the following conditions* [21]*:*

(1) $||\boldsymbol{O}(\boldsymbol{\xi})(t)|| \le M(a + \int_0^t ||\boldsymbol{\xi}(s)|| \mathrm{d}s)$
(2) $||\boldsymbol{O}(\boldsymbol{\xi})(t) - \boldsymbol{O}(\boldsymbol{\zeta})(t)|| \le M(\int_0^t ||\boldsymbol{\xi}(s) - \boldsymbol{\zeta}(s)|| \mathrm{d}s)$

In the above formula, if M and a are non-negative constants, then we can draw the following two conclusions:

(a) For $\boldsymbol{\zeta}(t) \in \boldsymbol{C}_r[0, T]$, there exists a unique $\boldsymbol{\xi}(t) \in \boldsymbol{C}_r[0, T]$, such that

$$\boldsymbol{\xi}(t) + \boldsymbol{O}(\boldsymbol{x})(t) = \boldsymbol{\zeta}(t) \tag{5}$$

(b) According to the definition of the function defined as $\overline{\boldsymbol{O}}(\boldsymbol{\zeta}) = \boldsymbol{O}(\boldsymbol{\xi})(t)$, where $\boldsymbol{\xi} \in \boldsymbol{C}_r[0, T]$ is the only solution defined by (a), there exists an $M_1 > 0$ such that

$$||\overline{\boldsymbol{O}}(\boldsymbol{\zeta})(t)|| \le M_1(a + \int_0^t ||\boldsymbol{\zeta}(s)|| \mathrm{d}s) \tag{6}$$

Lemma 3 . *Let the constant series $\{b_k\}_{k \ge 0}$, $b_k \ge 0$ converge to zero, and the function $\boldsymbol{O}_k(\boldsymbol{\theta})(t)$ satisfy* [21]

$$||\boldsymbol{O}_k(\boldsymbol{\theta})(t)|| = K(b_k + \int_0^t ||\boldsymbol{\theta}_k(\tau)|| \mathrm{d}\tau) \tag{7}$$

In the previous expression, $K > 1$ is a constant. If we assume that $\boldsymbol{\Psi}(t)$, which can be $r \times r$, is a dimensional matrix of continuous functions, and $\boldsymbol{\Psi} : \boldsymbol{C}_r[0, T] \to \boldsymbol{C}_r[0, T]$, then:

$$\boldsymbol{\Psi}(\boldsymbol{\theta})(t) = \boldsymbol{\Psi}(t)\boldsymbol{\theta}(t) \tag{8}$$

From the above equations, it follows that when the spectral radius of $\boldsymbol{\Psi} < 1$, then:

$$\lim_{k \to \infty} (\boldsymbol{\Psi} + \boldsymbol{O}_k)(\boldsymbol{\Psi} + \boldsymbol{O}_{k-1}) \cdots (\boldsymbol{\Psi} + \boldsymbol{O}_0)(\boldsymbol{\theta})(t) = 0 \tag{9}$$

3. Arbitrary Initial Value Suppression Strategy Based on Predefined-Time Convergence Sliding Mode Surface

3.1. Arbitrary Initial Value Suppression Strategy and Its Principle

According to the sliding mode control principle [22], a system state whose initial value is located at any position in the state space can reach and stabilize in the sliding mode surface $\boldsymbol{S}(\boldsymbol{x}(t))$ under the sliding mode controller and within the sliding mode surface (equivalent to $\boldsymbol{S}(\boldsymbol{x}(t)) \equiv 0$) sliding to the equilibrium point, O. The sliding mode control (SMC) law is illustrated in Figure 1.

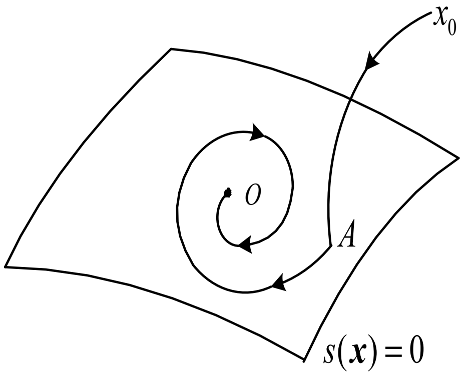

Figure 1. Schematic diagram of the principle, sliding mode control principle.

This study offers an arbitrary iterative initial value suppression control technique based on a predefined-time convergence sliding mode surface to tackle the arbitrary starting value issue in iterative learning control, using a control goal system (1) and the SMC principle.

When the starting state of the k-th iterative learning is at any point in space, it is the same as the initial value of trajectory or route tracking errors, $\boldsymbol{e}_k(0) \neq 0$. By applying the SMC concept, it is possible to build a sliding mode surface,$\boldsymbol{S}(\boldsymbol{e}_k(t))$,with predefined-time convergence characteristics and an iterative learning controller, $\boldsymbol{u}_k(t)$, allowing the controller, $\boldsymbol{u}_k(t)$,to drive errors,$\boldsymbol{e}_k(t)$, that arrive at any starting position and stabilize in the sliding mode surface (equivalent to $\boldsymbol{S}(\boldsymbol{e}_k(t)) \equiv 0$). When the tracking error $\boldsymbol{e}_k(t)$ is stabilized in the sliding mode surface according to the predefined-time convergence characteristics of the SMC surface, and when the tracking error, $\boldsymbol{e}_k(t)$, returns to zero within the predetermined period, T_s; that is, when $\boldsymbol{e}_k(t) \equiv 0, t \in [T_s, T]$, the goals of suppressing the issue of random starting values and achieving precise tracking of the intended trajectory are both realized.

To achieve the trajectory tracking error satisfying $\boldsymbol{e}_k(t) \equiv 0$ within the predefined-time interval $t \in [T_s, T]$, several core problems present themselves. The first is ensuring that the tracking error, $\boldsymbol{e}_k(t)$, converges and stabilizes within the sliding mode surface $\boldsymbol{S}(\boldsymbol{e}_k)$ after finite iterative learning, that is, $\lim_{k\to\infty} \boldsymbol{S}(\boldsymbol{e}_k(t)) = 0$. The second is that in the sliding mode surface $\boldsymbol{S}(\boldsymbol{e}_k(t))$, trajectory tracking error $\boldsymbol{e}_k(t)$ converges to the equilibrium point in a predefined-time, T_s, that is, $\lim_{t\to T_s} \boldsymbol{e}_k(t) = 0$. Based on the above two core problems, this paper designs a controller that suppresses the arbitrary iterative initial value problem in two steps. The first step is to design the sliding mode surface $\boldsymbol{S}(\boldsymbol{e}_k(t))$ with the characteristic of converging to the equilibrium point within the predefined-time, T_s, to ensure that the tracking error $\boldsymbol{e}_k(t)$ in sliding mode surface $\boldsymbol{S}(\boldsymbol{e}_k(t))$ converges to the equilibrium point within the predefined-time, T_s. The second step is to design an iterative learning controller to ensure the convergence of iterative learning, so that the tracking error $\boldsymbol{e}_k(t)$ reaches and stabilizes in the sliding surface $\boldsymbol{S}(\boldsymbol{e}_k(t))$.

3.2. Predefined-Time Convergence Lyapunov Stability Criterion and Sliding Mode Surface Design

Predefined-time convergence is the key to suppressing the arbitrary initial value of iteration and ensuring that the trajectory tracking error can achieve accurate tracking before the predefined-time, T_s. The definition of predefined-time stability is given below.

Definition 1. *For a second-order nonlinear system (1), if there is a preset constant, $T_s > 0$, such that for any $t \in [0, \infty]$ the condition is satisfied:*

$$\text{when} t \to T_s,\ \lim_{t\to T_s} \boldsymbol{y}(t) = 0;\ \text{When } t \geq T_s,\ \text{ithas } \boldsymbol{y}(t) \equiv 0$$

Then the second-order nonlinear system (1) is globally predefined-time stable.

A Lyapunov stability criterion of the predefined-time convergence is presented below and proven theoretically in order to facilitate the assessment of the nonlinear system's global predefined-time convergence.

Theorem 1. *In a nonlinear system (1), for any given predefined-time, $T_s > 0$, if there exsits a positively definite and radially unbounded Lyapunov function, $V(t)$, which satisfies*

$$\dot{V}(t) \leq -\frac{\pi}{2\lambda T_s\sqrt{ab}}(aV^{1-\lambda}(t) + bV^{1+\lambda}(t)) \tag{10}$$

where the parameters satisfy $0 < \lambda < 1, a > 0, b > 0$, then

(1) *if $V(0) \neq 0$, then the system is global predefined-time stable and converges to the equilibrium time: $t_s = \frac{2T_s}{\pi}\arctan\left(\sqrt{\frac{b}{a}}V^{\lambda}(0)\right) < T_s$ and*

(2) *f $V(0) = 0$, then $V(t) \equiv 0$, meaning that the system state is always at the equilibrium point.*

Proof . According to $\dot{V}(t) \leq -\frac{\pi}{2\lambda T_s\sqrt{ab}}(aV^{1-\lambda}(t) + bV^{1+\lambda}(t))$, adding the nonnegative constant $\Delta \geq 0$ to the right hand means that

$$\dot{V}(t) = -\frac{\pi}{2\lambda T_s\sqrt{ab}}(aV^{1-\lambda}(t) + bV^{1+\lambda}(t)) - \Delta$$

changes it format, giving:

$$\begin{aligned} \frac{\mathrm{d}V(t)}{\mathrm{d}t} &= -\frac{\pi}{2\lambda T_s\sqrt{ab}}aV^{1-\lambda}(t)(1 + \frac{b}{a}V^{2\lambda}(t)) - \Delta \\ &= -\frac{\pi}{2\lambda T_s\sqrt{ab}}aV^{1-\lambda}(t)(1 + \frac{b}{a}V^{2\lambda}(t) + \frac{2\lambda T_s\sqrt{ab}}{\pi aV^{1-\lambda}(t)}\Delta) \end{aligned} \tag{11}$$

After transforming (11), we obtain

$$\frac{\pi\sqrt{a}}{2T_s\sqrt{b}}\mathrm{d}t = -\frac{\lambda V^{\lambda-1}(t)\mathrm{d}V(t)}{1 + (\sqrt{\frac{b}{a}}V^{\lambda}(t))^2 + \frac{2\lambda T_s\sqrt{ab}}{\pi aV^{1-\lambda}(t)}\Delta} = -\frac{\mathrm{d}V^{\lambda}(t)}{1 + (\sqrt{\frac{b}{a}}V^{\lambda}(t))^2 + \frac{2\lambda T_s\sqrt{ab}}{\pi aV^{1-\lambda}(t)}\Delta} \tag{12}$$

After compiling differentiation (12), in order to make the integration process simpler, we have

$$\sqrt{\frac{b}{a}}\frac{\pi}{2T_s}\sqrt{\frac{a}{b}}\mathrm{d}t = \frac{\pi}{2T_s}\mathrm{d}t = -\frac{\mathrm{d}(\sqrt{\frac{b}{a}}V^{\lambda}(t))}{1 + (\sqrt{\frac{b}{a}}V^{\lambda}(t))^2 + \frac{2\lambda T_s\sqrt{ab}}{\pi aV^{1-\lambda}(t)}\Delta} \tag{13}$$

Assume that $V(t_s) = 0$ at time t_s and integrate both sides of Equation (13) simultaneously on $(0, t_s]$. Since $V(t) \geq 0$, $\Delta \geq 0$, then $\frac{2\lambda T_s\sqrt{ab}}{\pi aV^{1-\lambda}(t)}\Delta \geq 0$ and, according to $\lim_{t\to\infty}\arctan(t) = \frac{\pi}{2}$, we have:

$$\begin{aligned} &\int_0^{t_s}\frac{\pi}{2T_s}\mathrm{d}t = -\int_0^{t_s}\frac{\mathrm{d}(\sqrt{\frac{b}{a}}V^{\lambda}(t))}{1+(\sqrt{\frac{b}{a}}V^{\lambda}(t))^2 + \frac{2\lambda T_s\sqrt{ab}}{\pi aV^{1-\lambda}(t)}\Delta} = -\int_{V(0)}^{V(t_s)}\frac{\mathrm{d}(\sqrt{\frac{b}{a}}V^{\lambda}(t))}{1+(\sqrt{\frac{b}{a}}V^{\lambda}(t))^2 + \frac{2\lambda T_s\sqrt{ab}}{\pi aV^{1-\lambda}(t)}\Delta} \\ &= \int_0^{V(0)}\frac{\mathrm{d}(\sqrt{\frac{b}{a}}V^{\lambda}(t))}{1+(\sqrt{\frac{b}{a}}V^{\lambda}(t))^2 + \frac{2\lambda T_s\sqrt{ab}}{\pi aV^{1-\lambda}(t)}\Delta} \leq \int_0^{V(0)}\frac{\mathrm{d}(\sqrt{\frac{b}{a}}V^{\lambda}(t))}{1+(\sqrt{\frac{b}{a}}V^{\lambda}(t))^2} \\ &\Rightarrow \frac{\pi t}{2T_s}\Big|_0^{t_s} \leq \arctan\left(\sqrt{\frac{b}{a}}V^{\lambda}(t)\right)\Big|_0^{V(0)} \Rightarrow \frac{\pi}{2T_s}t_s \leq \arctan\left(\sqrt{\frac{b}{a}}V^{\lambda}(0)\right) \\ &\Rightarrow t_s \leq \frac{2T_s}{\pi}\arctan\left(\sqrt{\frac{b}{a}}V^{\lambda}(0)\right) \leq \frac{2T_s}{\pi}\frac{\pi}{2} = T_s \end{aligned} \tag{14}$$

When $V(0) = 0$, $t_s \leq \frac{2T_s}{\pi}\arctan\left(\sqrt{\frac{b}{a}}V^{\lambda}(0)\right) = 0$, it means that $V(t) \equiv 0$ is always true, indicating that the system state has always been at the equilibrium point. □

The predefined-time convergence stability criterion of the nonlinear system is mainly used to determine whether the trajectory tracking error located in the sliding mode surface can converge to the origin within a predefined time. We can derive a predefined-time sliding mode surface. The main goal of the proposed surface is to converge at that particular predefined time. This kind of time-varying sliding surface with predefined time characteristics can be described as follows:

$$S(e(t)) = \begin{cases} \dot{e}(t) + \frac{\pi}{2\lambda T_s\sqrt{ab}}(ae^{1-\lambda}(t) + be^{1+\lambda}(t)) - \\ \left(\dot{e}(0) + \frac{\pi}{2\lambda T_s\sqrt{ab}}(ae^{1-\lambda}(0) + be^{1+\lambda}(0))\right)\exp(-\alpha t) & t \leq t_\Delta \\ \dot{e}(t) + \frac{\pi}{2\lambda T_s\sqrt{ab}}(ae^{1-\lambda}(t) + be^{1+\lambda}(t)) & t_\Delta \leq t \leq T \end{cases} \tag{15}$$

where the parameters satisfy $0 < \lambda < 1, a > 0, b > 0, \alpha > 0$, T_s is the predefined-time, and t_Δ is a smaller constant that satisfies $t_\Delta \ll T_s$. Subsequently, a trajectory tracking error, $e(t)$, in the sliding mode surface may go to zero within a predefined-time T_s, which is given in the form of a theorem, and can be proved theoretically.

Theorem 2. *For any predefined-time $T_s > 0$, when the sliding mode surface (15) satisfies $S(e(t)) = 0$, it shows that the error $e(t)$ will converge to zero within the predefined-time, T_s, and the convergence time is:*

$$t_s = \frac{2T_s}{\pi}\arctan\left(\sqrt{\frac{\overline{b}}{\overline{a}}}V_1^{\overline{\lambda}}(0)\right) < T_s \tag{16}$$

where $\overline{a} = a2^{1-0.5\lambda}, \overline{b} = b2^{1+0.5\lambda}, \overline{\lambda} = 0.5\lambda$.

Proof. Note variable as:

$$\omega(t) = \left(\dot{e}(0) + \frac{\pi}{2\lambda T_s\sqrt{ab}}(ae^{1-\lambda}(0) + be^{1+\lambda}(0))\right)\exp(-\alpha t) \tag{17}$$

When $S(e(t)) = 0$ has $\dot{e}(t) = \begin{cases} -\frac{\pi}{2\lambda T_s\sqrt{ab}}(ae^{1-\lambda}(t) + be^{1+\lambda}(t)) + \omega(t) & t \leq t_\Delta \\ -\frac{\pi}{2\lambda T_s\sqrt{ab}}(ae^{1-\lambda}(t) + be^{1+\lambda}(t)) & t_\Delta \leq t \leq T \end{cases}$' establish the Lyapunov function as $V_1(t) = 0.5e^T(t)e(t)$, and derive it to get

$$\begin{aligned} \dot{V}_1(t) &= e^T(t)\dot{e}(t) \\ &= \begin{cases} -\frac{\pi}{2\lambda T_s\sqrt{ab}}(ae^T(t)e^{1-\lambda}(t) + be^T(t)e^{1+\lambda}(t)) + e^T(t)\omega(t) & t \leq t_\Delta \\ -\frac{\pi}{2\lambda T_s\sqrt{ab}}(ae^T(t)e^{1-\lambda}(t) + be^T(t)e^{1+\lambda}(t)) & t_\Delta \leq t \leq T \end{cases} \\ &= \begin{cases} -\frac{\pi}{2\lambda T_s\sqrt{ab}}(a2^{1-0.5\lambda}{V_1}^{1-0.5\lambda}(t) + b2^{1+0.5\lambda}{V_1}^{1+0.5\lambda}(t)) + e^T(t)\omega(t) & t \leq t_\Delta \\ -\frac{\pi}{2\lambda T_s\sqrt{ab}}(a2^{1-0.5\lambda}{V_1}^{1-0.5\lambda}(t) + b2^{1+0.5\lambda}{V_1}^{1+0.5\lambda}(t)) & t_\Delta \leq t \leq T \end{cases} \\ &= \begin{cases} -\frac{\pi}{2\overline{\lambda} T_s\sqrt{\overline{a}\overline{b}}}(\overline{a}{V_1}^{1-\overline{\lambda}}(t) + \overline{b}{V_1}^{1+\overline{\lambda}}(t)) + e^T(t)\omega(t) & t \leq t_\Delta \\ -\frac{\pi}{2\overline{\lambda} T_s\sqrt{\overline{a}\overline{b}}}(\overline{a}{V_1}^{1-\overline{\lambda}}(t) + \overline{b}{V_1}^{1+\overline{\lambda}}(t)) & t_\Delta \leq t \leq T \end{cases} \end{aligned} \tag{18}$$

where $\overline{a} = a2^{1-0.5\lambda}, \overline{b} = b2^{1+0.5\lambda}, \overline{\lambda} = 0.5\lambda$. Due to the fact that t_Δ satisfies $t_\Delta \ll T_s$, and $\omega(t)$ contains the exponent part $\exp(-\alpha t)$, when α takes a larger value, $\omega(t)$ can quickly tend to zero, so the influence of $\boldsymbol{e}^T(t)\omega(t)$ on $\dot{V}_1(t)$ is very small and (18) can actually be equivalent to:

$$\dot{V}_1(t) = -\frac{\pi}{2\overline{\lambda}T_s\sqrt{\overline{a}\overline{b}}}(\overline{a}V_1{}^{1-\overline{\lambda}}(t) + \overline{b}V_1{}^{1+\overline{\lambda}}(t)) \tag{19}$$

According to Theorem 1, when the sliding mode surface (15) fulfils the condition $\boldsymbol{S}(\boldsymbol{e}(t)) = 0$, which implies an error, $\boldsymbol{e}(t)$, it may converge to zero within the predefined-time, with a convergence time of

$$t_s = \frac{2T_s}{\pi}\arctan\left(\sqrt{\frac{\overline{b}}{\overline{a}}}V_1^{\overline{\lambda}}(0)\right) < T_s$$

From the above analysis, it can be seen that the predefined-time convergence sliding mode iterative learning control strategy proposed in this paper can be described as designing an iterative learning controller for a sliding mode surface, $\boldsymbol{S}(\boldsymbol{e}_k(t))$, so that the sliding mode surface $\boldsymbol{S}(\boldsymbol{e}_k(t))$ converges to 0 after iterative learning. This means that the trajectory tracking error, $\boldsymbol{e}_k(t)$, will converge to 0 within the predefined-time, T_s, achieving the control purpose of accurately tracking the desired trajectory within the preset interval $[T_s, T]$. □

This control strategy transforms the trajectory tracking control problem with an initial trajectory tracking error value that is not zero into a sliding mode surface tracking control problem with the initial value of the sliding surface at zero. It also establishes a bridge connecting the iterative learning control theory of arbitrary iterative initial value and the same iterative initial value. The theoretical connecting bridge not only solves the arbitrary initial value problem of iteration but also simplifies the theoretical proof of the convergence of iterative learning and can take advantage of the existing theoretical achievements of iterative learning control.

For convenience, in the theoretical proof of the convergence of iterative learning, we will take $\boldsymbol{S}(\boldsymbol{e}_k(t))$ as $\boldsymbol{S}_k(t)$.

4. Convergence Analysis of PD-Type ILC

To ensure that the output of the nonlinear system accurately tracks the desired trajectory within the predefined-time interval, $[T_s, T]$, the sliding mode surface $\boldsymbol{S}_k(t)$ must converge to zero. Therefore, let the desired trajectory of the sliding mode surface $\boldsymbol{S}_k(t)$ be $\mathrm{S_d}(t) = 0$, and $\dot{\mathrm{S}}_\mathrm{d}(t) = 0$, then denote the tracking error $\delta\boldsymbol{S}_k(t) = \boldsymbol{S}_k(t) - \boldsymbol{S}_\mathrm{d}(t) = \boldsymbol{S}_k(t)$.

The PD-type closed-loop iterative learning controller for sliding mode surfaces is designed as:

$$\begin{aligned} \boldsymbol{u}_{k+1}(t) &= \boldsymbol{u}_k(t) + \boldsymbol{H}_1\delta\boldsymbol{S}_{k+1}(t) + \boldsymbol{H}_2\delta\dot{\boldsymbol{S}}_{k+1}(t) \\ &= \boldsymbol{u}_k(t) + \boldsymbol{H}_1\boldsymbol{S}_{k+1}(t) + \boldsymbol{H}_2\dot{\boldsymbol{S}}_{k+1}(t) \end{aligned} \tag{20}$$

where $\boldsymbol{H}_1$ is proportional gain matrix, $\boldsymbol{H}_2$ is differential gain matrix, and $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ are positive-definite matrixes.

Theorem 3. *In relation to the second-order nonlinear system defined in (1), if the PD-type iterative learning controller is the controller which is described in (19), and the spectral radius satisfies*

$$\rho([\boldsymbol{I} - \boldsymbol{B}\boldsymbol{H}_2]^{-1}) < 1 \tag{21}$$

then, the sliding mode surface $\boldsymbol{S}_k(t)$ will converge to zero under the condition of $k \to \infty$, ie $\lim_{k\to\infty} \boldsymbol{S}_k(t) = 0$.

Proof. Introduce variable as:

$$\begin{aligned}
\boldsymbol{r}(t) &= -\dot{\boldsymbol{y}}_{\mathrm{d}}(t) + \begin{cases} \frac{\pi}{2\lambda T_s\sqrt{ab}}(a\boldsymbol{e}^{1-\lambda}(t) + b\boldsymbol{e}^{1+\lambda}(t)) - \omega(t) & t \le t_\Delta \\ \frac{\pi}{2\lambda T_s\sqrt{ab}}(a\boldsymbol{e}^{1-\lambda}(t) + b\boldsymbol{e}^{1+\lambda}(t)) & t_\Delta \le t \le T \end{cases} \\
\dot{\boldsymbol{r}}(t) &= -\ddot{\boldsymbol{y}}_{\mathrm{d}}(t) + \begin{cases} \frac{\pi}{2\lambda T_s\sqrt{ab}}(a(1-\lambda)\boldsymbol{e}^{-\lambda}(t) + b(1+\lambda)\boldsymbol{e}^{\lambda}(t))\dot{\boldsymbol{e}}(t) + \alpha\omega(t) & t \le t_\Delta \\ \frac{\pi}{2\lambda T_s\sqrt{ab}}(a(1-\lambda)\boldsymbol{e}^{-\lambda}(t) + b(1+\lambda)\boldsymbol{e}^{\lambda}(t))\dot{\boldsymbol{e}}(t) & t_\Delta \le t \le T \end{cases}
\end{aligned} \tag{22}$$

Then

$$\begin{aligned} \boldsymbol{S}(t) &= \dot{\boldsymbol{y}}(t) + \boldsymbol{r}(t) \\ \dot{\boldsymbol{S}}(t) &= \ddot{\boldsymbol{y}}(t) + \dot{\boldsymbol{r}}(t) \end{aligned}$$

According to Formula (1), we have

$$\begin{aligned} \boldsymbol{S}_k(t) &= \dot{\boldsymbol{x}}_{1k} + \boldsymbol{r}(t) = \boldsymbol{x}_{2k} + \boldsymbol{r}_k(t) \\ \dot{\boldsymbol{S}}_k(t) &= \dot{\boldsymbol{x}}_{2k} + \dot{\boldsymbol{r}}_k(t) = \boldsymbol{f}(\boldsymbol{x}_k, t) + \boldsymbol{B}\boldsymbol{u}_k + \dot{\boldsymbol{r}}_k(t) \triangleq F(\boldsymbol{S}_k, t) + \boldsymbol{B}\boldsymbol{u}_k \end{aligned} \tag{23}$$

where $F(\boldsymbol{S}_k, t) = \boldsymbol{f}(\boldsymbol{x}_k, t) + \dot{\boldsymbol{r}}_k(t)$. Because $\boldsymbol{f}(\boldsymbol{x}_k(t), t)$ is a function of $\boldsymbol{x}_{1k}, \boldsymbol{x}_{2k}$, and within the time interval $t \in [0, T]$, it satisfies the Lipschitz condition regarding $\boldsymbol{x}_k(t)$ and $\boldsymbol{S}_k(t) = \boldsymbol{x}_{2k} + \boldsymbol{r}_k(t)$. Therefore, the function $F(\boldsymbol{S}_k, t)$ also satisfies the Lipschitz condition with variable $\boldsymbol{S}_k$. That is, there is a constant $M_2 > 0$, with

$$||\boldsymbol{F}(\boldsymbol{S}_k, t) - \boldsymbol{F}(\boldsymbol{S}_{\mathrm{d}}, t)|| \le M_2||\boldsymbol{S}_k - \boldsymbol{S}_{\mathrm{d}}|| \tag{24}$$

The unique solution of this differential equation is obtained according to $\dot{\boldsymbol{S}}_k = \boldsymbol{F}(\boldsymbol{S}_k, t) + \boldsymbol{B}\boldsymbol{u}_k$ add $\boldsymbol{S}_k(0) = 0$.

$$\boldsymbol{S}_k = \int_0^t [\boldsymbol{F}(\boldsymbol{S}_k) + \boldsymbol{B}\boldsymbol{u}_k] d\tau = \int_0^t \boldsymbol{F}(\boldsymbol{S}_k) d\tau + \int_0^t \boldsymbol{B}\boldsymbol{u}_k d\tau \tag{25}$$

Then:

$$\begin{aligned} &\boldsymbol{S}_{k+1}(t) = \textstyle\int_0^t \boldsymbol{F}(\boldsymbol{S}_{k+1})\mathrm{d}\tau + \int_0^t \boldsymbol{B}\boldsymbol{u}_{k+1}\mathrm{d}\tau = \\ &\textstyle\int_0^t \boldsymbol{F}(\boldsymbol{S}_{k+1}, \tau)\mathrm{d}\tau + \int_0^t \boldsymbol{B}\boldsymbol{u}_k\mathrm{d}\tau + \int_0^t \boldsymbol{B}[\boldsymbol{H}_1\boldsymbol{S}_{k+1}(\tau) + \boldsymbol{H}_2\dot{\boldsymbol{S}}_{k+1}(\tau)]\mathrm{d}\tau = \\ &\textstyle\int_0^t [\boldsymbol{F}(\boldsymbol{S}_{k+1}) - \boldsymbol{F}(\boldsymbol{S}_k)]\mathrm{d}\tau + \boldsymbol{S}_k + \int_0^t \boldsymbol{B}\boldsymbol{H}_1\boldsymbol{S}_{k+1}(\tau)\mathrm{d}\tau + \boldsymbol{B}\boldsymbol{H}_2\boldsymbol{S}_{k+1} - \int_0^t \boldsymbol{S}_{k+1}(\tau)\frac{\mathrm{d}\boldsymbol{B}\boldsymbol{H}_2}{\mathrm{d}\tau}\mathrm{d}\tau \end{aligned} \tag{26}$$

Then

$$\Big[\boldsymbol{I} - \boldsymbol{B}\boldsymbol{H}_2]\boldsymbol{S}_{k+1}(t) = \boldsymbol{S}_k(t) + \int_0^t [\boldsymbol{F}(\boldsymbol{S}_{k+1}, \tau) - \boldsymbol{F}(\boldsymbol{S}_k, \tau)]\mathrm{d}\tau + \int_0^t \boldsymbol{B}\boldsymbol{H}_1\boldsymbol{S}_{k+1}\mathrm{d}\tau - \int_0^t \boldsymbol{S}_{k+1}\frac{\mathrm{d}\boldsymbol{B}\boldsymbol{H}_2}{\mathrm{d}\tau}\mathrm{d}\tau \tag{27}$$

is deformed to obtain

$$\boldsymbol{S}_{k+1} = [\boldsymbol{I} - \boldsymbol{B}\boldsymbol{H}_2]^{-1}\{\boldsymbol{S}_k + \int_0^t [\boldsymbol{F}(\boldsymbol{S}_{k+1}) - \boldsymbol{F}(\boldsymbol{S}_k)]\mathrm{d}\tau + \int_0^t \boldsymbol{B}\boldsymbol{H}_1\boldsymbol{S}_{k+1}\mathrm{d}\tau - \int_0^t \boldsymbol{S}_{k+1}\frac{\mathrm{d}\boldsymbol{B}\boldsymbol{H}_2}{\mathrm{d}\tau}\mathrm{d}\tau\} \tag{28}$$

Note $\boldsymbol{\Psi}(t) = [\boldsymbol{I} - \boldsymbol{B}\boldsymbol{H}_2]^{-1}$, and is defined as

$$\boldsymbol{K}_{k+1}(\boldsymbol{S}_{k+1})(t) = -[\boldsymbol{I} - \boldsymbol{B}\boldsymbol{H}_2]^{-1}\{\int_0^t [\boldsymbol{F}(\boldsymbol{S}_{k+1}) - \boldsymbol{F}(\boldsymbol{S}_k)]\mathrm{d}\tau + \int_0^t \boldsymbol{B}\boldsymbol{H}_1\boldsymbol{S}_{k+1}\mathrm{d}\tau - \int_0^t \boldsymbol{S}_{k+1}\frac{\mathrm{d}\boldsymbol{B}\boldsymbol{H}_2}{\mathrm{d}\tau}\mathrm{d}\tau\} \tag{29}$$

Then:

$$\boldsymbol{S}_{k+1}(t) + \boldsymbol{K}_{k+1}(\boldsymbol{S}_{k+1})(t) = \boldsymbol{\Psi}(t)\boldsymbol{S}_k(t) \tag{30}$$

According to Formula (25), it can be deduced that:

$$\begin{aligned} &\boldsymbol{S}_{k+1}(t) - \boldsymbol{S}_k(t) = \\ &\int_0^t [\boldsymbol{F}(\boldsymbol{S}_{k+1}) - \boldsymbol{F}(\boldsymbol{S}_k)]\mathrm{d}\tau + \int_0^t \boldsymbol{B}\boldsymbol{H}_1\boldsymbol{S}_{k+1}\mathrm{d}\tau + \boldsymbol{B}\boldsymbol{H}_2\boldsymbol{S}_{k+1} - \int_0^t \boldsymbol{S}_{k+1}\frac{\mathrm{d}\boldsymbol{B}\boldsymbol{H}_2}{\mathrm{d}\tau}\mathrm{d}\tau \end{aligned} \tag{31}$$

By taking the norm on both sides of (31) we obtain

$$\begin{aligned} &||\boldsymbol{S}_{k+1}(t) - \boldsymbol{S}_k(t)|| \le M_2\int_0^t ||\boldsymbol{S}_{k+1} - \boldsymbol{S}_k||\mathrm{d}\tau + \\ &\int_0^t ||\boldsymbol{B}\boldsymbol{H}_1\boldsymbol{S}_{k+1}||\mathrm{d}\tau + ||\boldsymbol{B}\boldsymbol{H}_2\boldsymbol{S}_{k+1}|| + \int_0^t ||\boldsymbol{S}_{k+1}||||\frac{\mathrm{d}\boldsymbol{B}\boldsymbol{H}_2}{\mathrm{d}\tau}||\mathrm{d}\tau \le \\ &M_2\int_0^t ||\boldsymbol{S}_{k+1} - \boldsymbol{S}_k||\mathrm{d}\tau + N_1\int_0^t ||\boldsymbol{S}_{k+1}||\mathrm{d}\tau + N_2||\boldsymbol{S}_{k+1}|| + N_3\int_0^t ||\boldsymbol{S}_{k+1}||\mathrm{d}\tau \le \\ &M_2\int_0^t ||\boldsymbol{S}_{k+1} - \boldsymbol{S}_k||\mathrm{d}\tau + N_4\int_0^t ||\boldsymbol{S}_{k+1}||\mathrm{d}\tau + N_2||\boldsymbol{S}_{k+1}|| \end{aligned} \tag{32}$$

where: $N_1 = \sup\limits_{\tau\in[0,T]} ||\boldsymbol{B}(\tau)\boldsymbol{H}_1||$, $N_2 = \sup\limits_{\tau\in[0,T]} ||\boldsymbol{B}(t)\boldsymbol{H}_2||$, $N_3 = \sup\limits_{\tau\in[0,T]} ||\frac{\mathrm{d}\boldsymbol{B}(\tau)\boldsymbol{H}_2}{\mathrm{d}\tau}||$, $N_4 = N_1 + N_3$

According to Lemma 1, we have:

$$\begin{aligned} &||\boldsymbol{S}_{k+1}(t) - \boldsymbol{S}_k(t)|| \le \\ &N_4\int_0^t ||\boldsymbol{S}_{k+1}||\mathrm{d}\tau + N_2||\boldsymbol{S}_{k+1}|| + M_2\int_0^t [N_4\int_0^\tau ||\boldsymbol{S}_{k+1}||\mathrm{d}v + N_2||\boldsymbol{S}_{k+1}||]e^{\mathrm{M}(t-\tau)}\mathrm{d}\tau \le \\ &N_5\int_0^t ||\boldsymbol{S}_{k+1}||\mathrm{d}\tau + N_2||\boldsymbol{S}_{k+1}|| \end{aligned} \tag{33}$$

where $N_5 = N_4 + M_2N_4T\mathrm{e}^{MT} + M_2N_2\mathrm{e}^{MT}$. By taking the norm on both sides of Equation (28), we obtain:

$$||\boldsymbol{K}_{k+1}(\boldsymbol{S}_{k+1})(t)|| \le ||[\boldsymbol{I} - \boldsymbol{B}\boldsymbol{H}_2]^{-1}||(M_2\int_0^t ||\boldsymbol{S}_{k+1} - \boldsymbol{S}_k||\mathrm{d}\tau + N_4\int_0^t ||\boldsymbol{S}_{k+1}||\mathrm{d}\tau) \le N_6\int_0^t ||\boldsymbol{S}_{k+1}||\mathrm{d}\tau \tag{34}$$

where $N_6 = \sup\limits_{t\in[0,T]} ||[\boldsymbol{I} - \boldsymbol{B}\boldsymbol{H}_2]^{-1}(M_2N_5T + M_2N_2N_4)$.

Note $N_7 = \max\{N_6, 1\}$, has

$$||\boldsymbol{K}_{k+1}(\boldsymbol{S}_{k+1})(t)|| \le N_7\int_0^t ||\boldsymbol{S}_{k+1}(\tau)||\mathrm{d}\tau \tag{35}$$

Similarly, it can be deduced that:

$$||\boldsymbol{K}_{k+1}(\boldsymbol{S}_{k+1})(t) - \boldsymbol{K}_{k+1}(\boldsymbol{S}_k)(t)|| \le N_8\int_0^t ||\boldsymbol{S}_{k+1}(\tau) - \boldsymbol{S}_k(\tau)||\mathrm{d}\tau \tag{36}$$

where the constant $N_8 > 1$. According to Lemma 2, the particular function designed as $\overline{\mathbf{K}}_{k+1}$, such that

$$\boldsymbol{S}_{k+1}(t) + \overline{\mathbf{K}}_{k+1}(\boldsymbol{P}\boldsymbol{S}_k)(t) = \boldsymbol{\Psi}(t)\boldsymbol{S}_k(t) \tag{37}$$

where $\overline{\mathbf{K}}_{k+1}(\boldsymbol{\Psi}\boldsymbol{S}_k)(t)$ satisfies

$$||\overline{\mathbf{K}}_{k+1}(\boldsymbol{\Psi}\boldsymbol{S}_k)(t)|| \le N_9\int_0^t ||\boldsymbol{\Psi}(\tau)\boldsymbol{S}_k(\tau)||d\tau \tag{38}$$

$N_9 > 0$ is a constant in the Formula (38).

Defining the function $\boldsymbol{O}_{k+1}$ is $\boldsymbol{O}_{k+1}(\boldsymbol{S}_k)(t) = -\overline{\mathbf{K}}_{k+1}(\boldsymbol{\Psi}\boldsymbol{S}_k)(t)$; according to Lemma 3, we know that $N_{10} > 1$ such that $||\boldsymbol{O}_{k+1}(\boldsymbol{S}_k)|| \le N_{10}\int_0^t ||\boldsymbol{\Psi}\boldsymbol{S}_k||\mathrm{d}\tau$, and

$$\boldsymbol{S}_{k+1}(t) = (\boldsymbol{\Psi}\boldsymbol{S}_k)(t) + \boldsymbol{O}_{k+1}(\boldsymbol{S}_k)(t) = (\boldsymbol{\Psi} + \boldsymbol{O}_{k+1})(\boldsymbol{S}_k)(t) = (\boldsymbol{\Psi} + \boldsymbol{O}_{k+1})\cdots(\boldsymbol{\Psi} + \boldsymbol{O}_1)(\boldsymbol{S}_0)(t) \tag{39}$$

As mentioned in Lemma 3, if the condition of a spectral radius of $\boldsymbol{\Psi}$ that is $\rho(\boldsymbol{\Psi}) = \rho([\boldsymbol{I} - \boldsymbol{B}\boldsymbol{H}_2]^{-1}) < 1$, then $\boldsymbol{S}_k(t) \to 0$, $k \to \infty$ is uniformly established for t, such that the sliding mode surface, $\boldsymbol{S}_k(t)$, converges to zero uniformly under the control of PD-type closed-loop iterative learning law.□

According tso Theorem 2, when $\boldsymbol{S}(\boldsymbol{e}_k(t)) = 0$, which signifies that tracking error, $\boldsymbol{e}_k(t)$, definitely converges to the equilibrium point within a predefined-time, T_s, then $\lim_{t\to T_s} \boldsymbol{e}_k(t) = 0$, such that the trajectory tracking error $\boldsymbol{e}_k(t)$ of the nonlinear system under arbitrary initial value of iteration is convergent in the predefined-time that is our control target.

5. Simulation Experiment

The simulation target of this article is a two degree of freedom (2-DOF) manipulator undergoing repeated motion within the control of a trajectory tracking controller, as shown below. The manipulator's dynamic model is described as follows (see below Figure 2):

$$\boldsymbol{D}(\boldsymbol{q}(t))\begin{bmatrix}\ddot{q}_1(t)\\ \ddot{q}_2(t)\end{bmatrix} + \boldsymbol{C}(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t))\begin{bmatrix}\dot{q}_1(t)\\ \dot{q}_2(t)\end{bmatrix} + \boldsymbol{g}(\boldsymbol{q}(t)) = \begin{bmatrix}u_1(t)\\ u_2(t)\end{bmatrix} \tag{40}$$

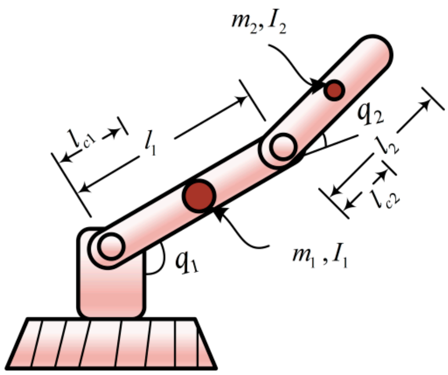

Figure 2. Two degree of freedom manipulator.

In the above-mentioned Equation (39), $\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \ddot{\boldsymbol{q}}(t)$ denote the joint position, velocity, and acceleration of the manipulator respectively. $\boldsymbol{D}(\boldsymbol{q}(t))$ denotes the inertial matrix, $\boldsymbol{C}(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t))$ signifies the centripetal force matrix, $\boldsymbol{g}(\boldsymbol{q}(t))$ shows the gravity vector, and $\boldsymbol{u}(t)$ denotes the control input. The expressions of the elements in the matrices $\boldsymbol{D}(\boldsymbol{q}(t))$, $\boldsymbol{C}(\boldsymbol{q}(t),\dot{\boldsymbol{q}}(t))$, and $\boldsymbol{g}(\boldsymbol{q}(t))$ are: $D_{11}(t) = m_1 l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos(q_2(t))) + I_1 + I_2$, $D_{12}(t) = m_2(l_{c2}^2 + l_1 l_{c2}\cos(q_2(t)) + I_2)$, $h(t) = -m_2 l_1 l_2 \sin(q_2(t))$, $D_{21}(t) = m_2(l_{c2}^2 + l_1 l_{c2} \cos(q_1(t) + q_2(t)))$, $D_{22}(t) = m_2 l_{c2}^2 + I_2$, $C_{11}(t) = h(t)q_2(t)$, $C_{12}(t) = h(t)q_1(t) + h(t)q_2(t)$, $C_{21}(t) = -h(t)q_1(t)$, $C_{22}(t) = 0$, $g_1(t) = (m_1 l_{c1} + m_2 l_1)g\cos(q_1(t)) + m_2 l_{c2} g\cos(q_1(t) + q_2(t))$, $g_2(t) = m_2 l_{c2} g\cos(q_1(t) + q_2(t))$

The relevant parameters of the manipulator are listed in Table 1.

Table 1. Parameters of robotic manipulator.

Parameter	Value	Unit
Manipulator's mass (m_1)	10	kg
Manipulator's mass (m_2)	5	kg
Length of joint (l_1)	1.00	m
Length of joint (l_2)	0.50	m
Length of joint (l_{c1})	0.50	m
Length of joint (l_{c2})	0.25	m
Moment of inertia (I_1)	0.83	kg.m^2
Moment of inertia (I_2)	0.30	kg.m^2
Gravitational acceleration (g)	9.81	m·s^{-2}

The simulation time was set to 20 s, the number of iterations to 10, and the predefined convergence time to $T_s = 8$s. With the Matlab command x_0 = (rand (4, 1)−0.5).*10, we produced a random initial point for each iteration, which we then used to determine the initial value of the position and velocity of iterative learning.

The controller parameters for the simulation investigation are given in Table 2.

Table 2. Controller parameters.

Element	Description
Simulation interval (t)	30 s
No. of iterations (K)	10 times
Predefined-convergence time (T_s)	8 s
Sliding surface	$p = 7$, $a = 1.4$, $b = 1$, $\alpha = 100$
ILC controller	PD Controller Proportional gain = $[1500, 0; 0, 800]$ Differential gain = $[1500, 0; 0, 400]$
Reference position signal	y_{1d} = sin(3t) y_{2d} = cos(3t)

5.1. Case 1: Control Performance (Without Disturbances)

Based on the PD-type closed-loop iterative learning controller designed in this paper for numerical simulation, the trajectory tracking results of the first, third, fifth, seventh, and tenth iterations are shown in Figures 3–7. Because the initial value of the iteration in each iterative learning process is randomly generated, the average value of the absolute value of the trajectory tracking error after the predefined time $T_s = 8$s (called the average absolute error) was selected as the iterative convergence evaluation standard. The iterative convergence graph of the absolute average error and the number of iterations is shown in Figure 8.

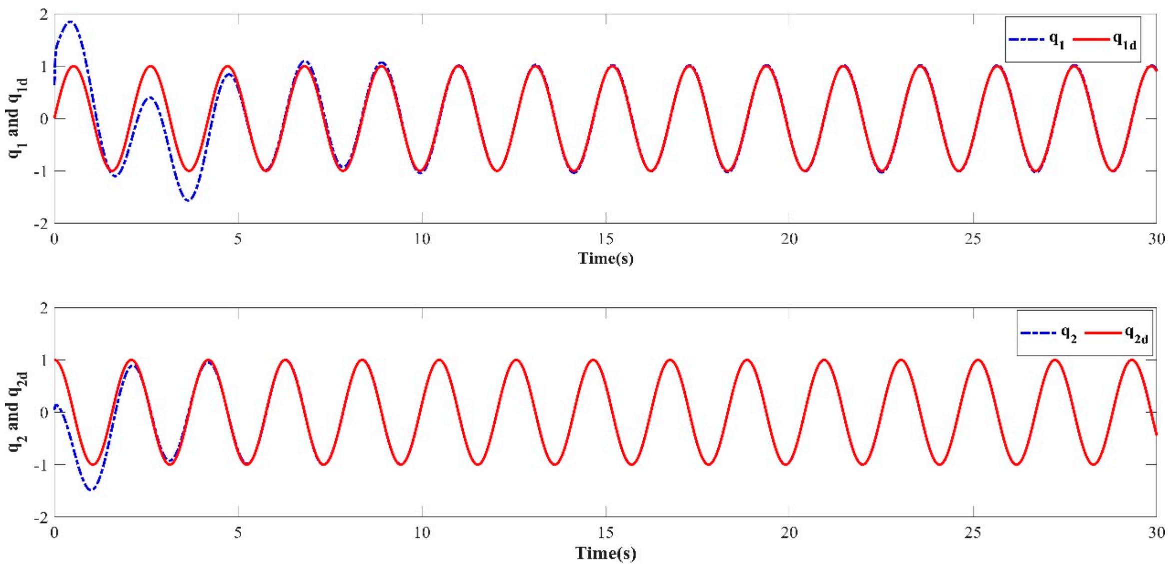

Figure 3. Learning trajectory in the first iteration.

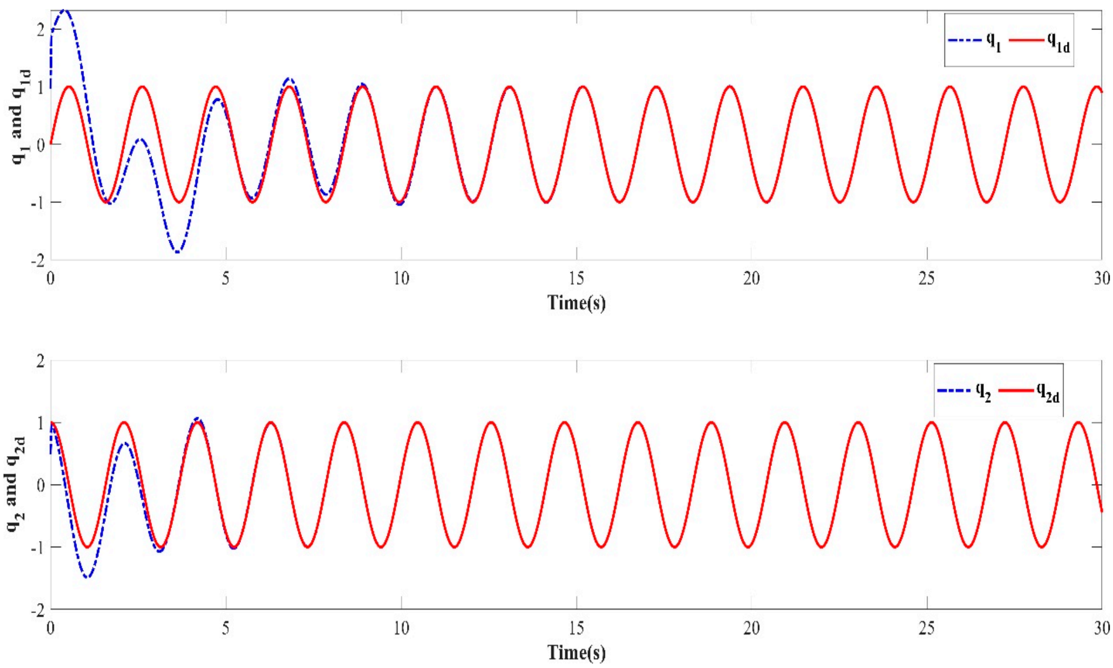

Figure 4. Learning trajectory in the third iteration.

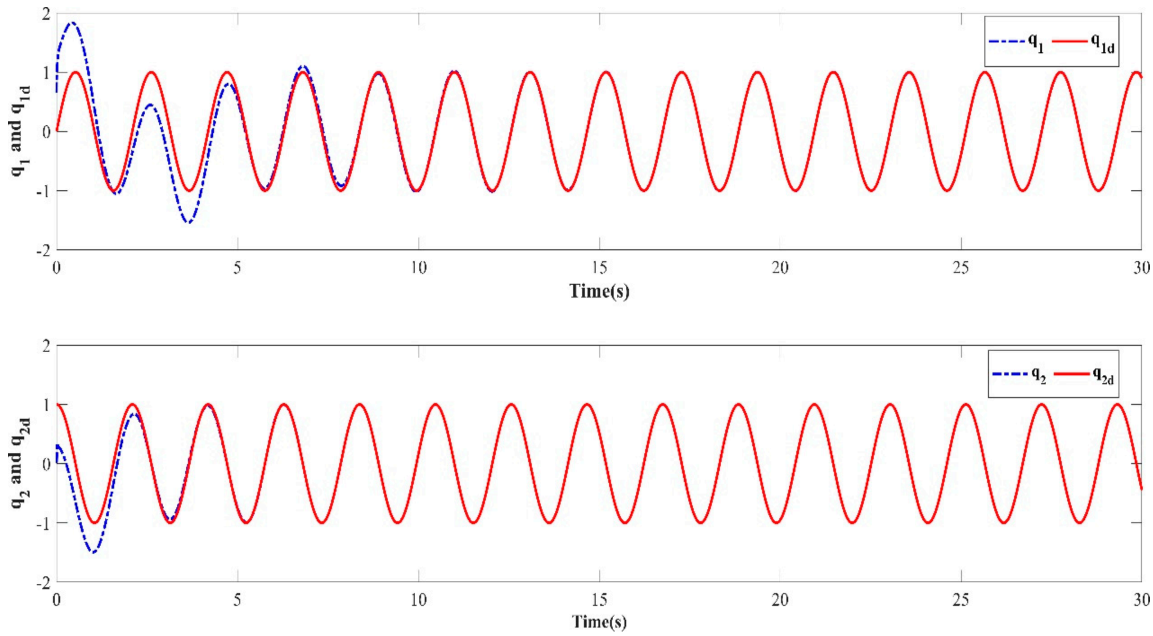

Figure 5. Learning trajectory in the fifth iteration.

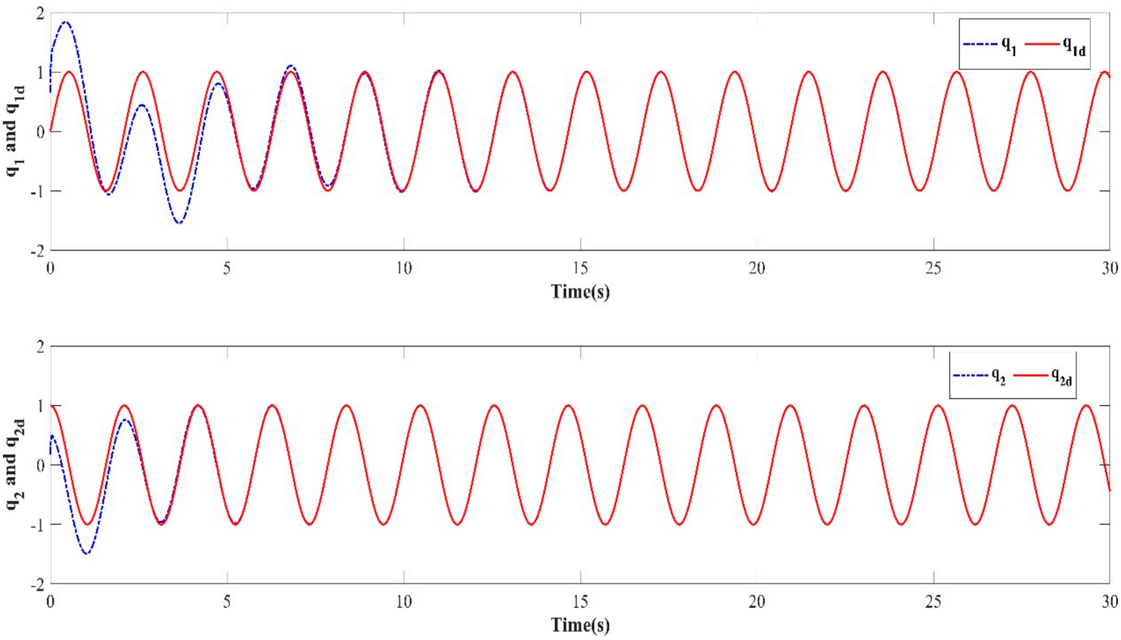

Figure 6. Learning trajectory in the seventh iteration.

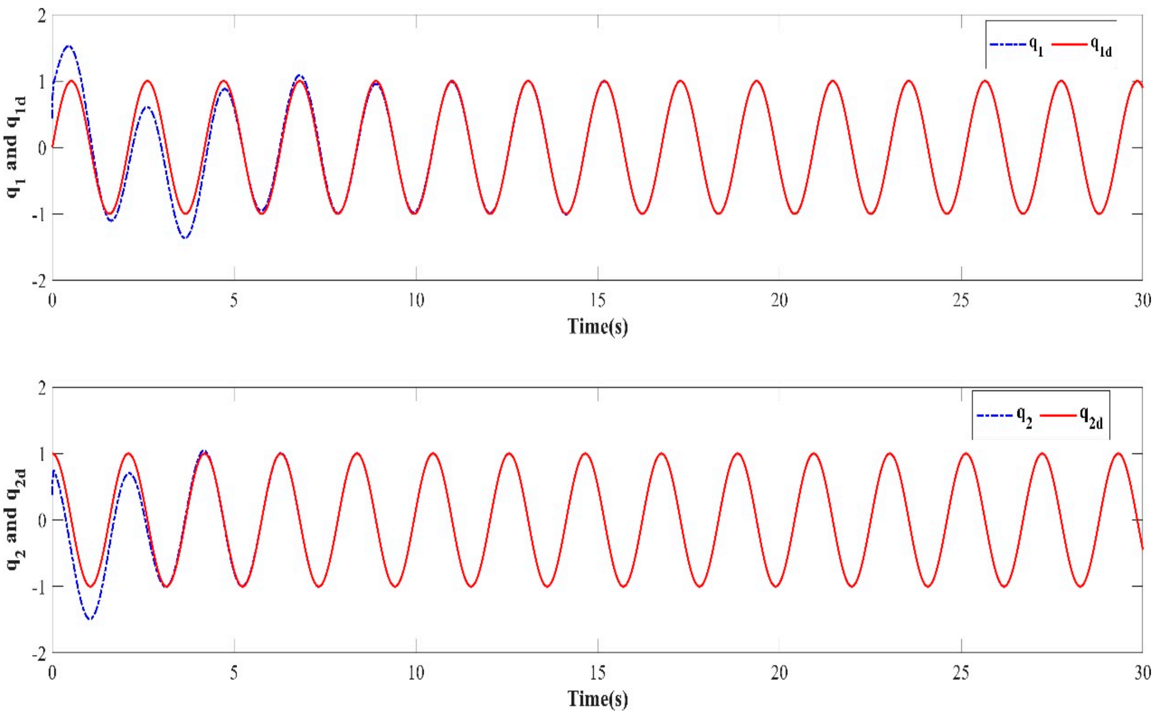

Figure 7. Learning trajectory in the tenth iteration.

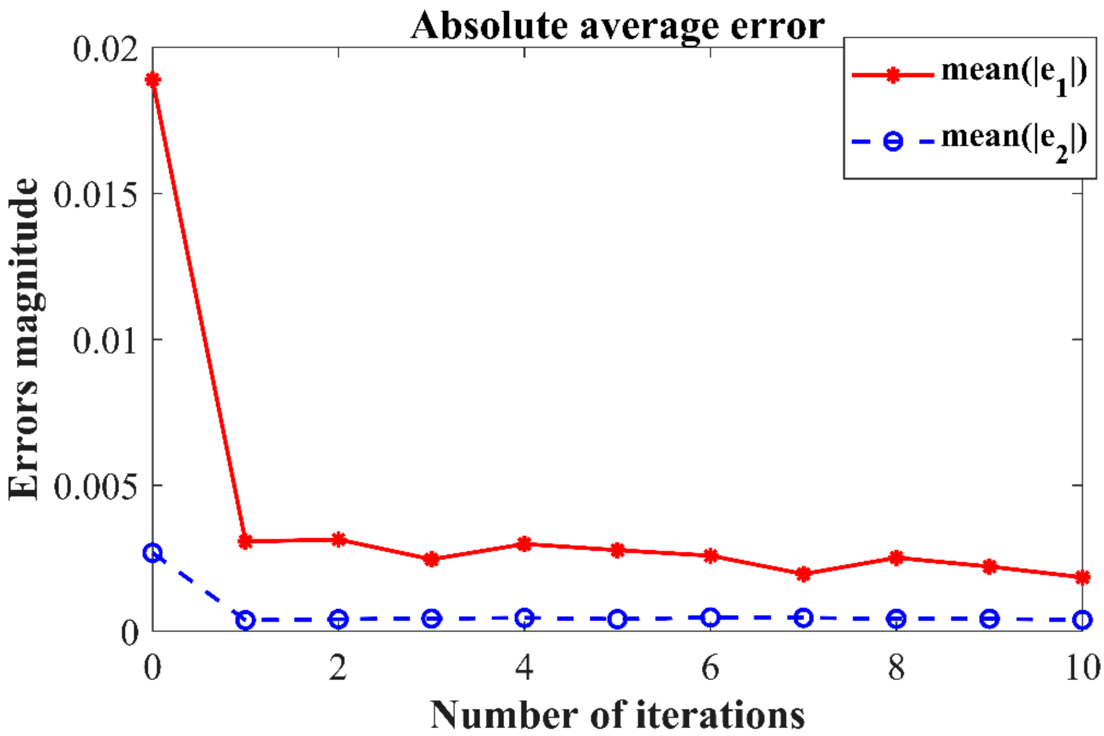

Figure 8. Iterative error convergence trend.

The numerical simulation results show that in the simulation environment, where the initial values $q_k(0)$ and $\dot{q}_k(0)$ were randomly generated in each iteration, based on the PD-type closed-loop iterative learning controller designed in this paper, the two degree of freedom manipulator system achieved iterative convergence after only two learning iterations. The absolute average error after 8s was lower than 0.004, which shows that, based on the control algorithm proposed in this paper, the end position of the manipulator can accurately track the desired trajectory after a predefined time, which verifies the feasibility of the algorithm. It effectively resolves the control problem such that the output of the nonlinear system can track the desired trajectory with high precision within the predefined time interval under an arbitrary initial value of iteration.

It can be seen from Figure 9 that the control torque of the manipulator was very large in the first few seconds, but that it then reduced to less than 60 N, while the value of u2 was even less than 30 N. This demonstrates that the proposed controller has smaller control input, which means the controller requires less effort from the manipulator.

5.2. Case 2: Robustness (With Disturbances)

In order to verify the robustness of the controller to external disturbances while keeping the control parameters the same as in Case 1 and the initial value of the iteration in each iterative learning process randomly generated, a numerical simulation was performed after adding an external disturbance $\boldsymbol{d}(t) = [3sin(t), 1(1 - e^{-t})]^T$ to the dynamic system of the two degree of freedom manipulator. The trajectory tracking effect after 10 iterations is shown in Figure 10. The iterative convergence diagram of the mean absolute error with respect to the number of iterations is shown in Figure 11.

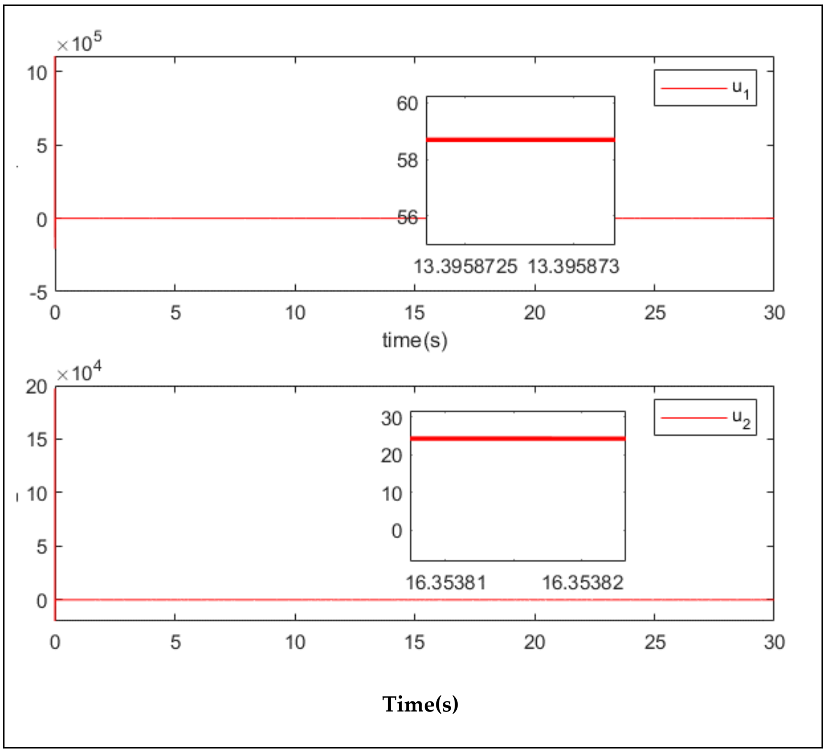

Figure 9. Control input in the tenth iteration.

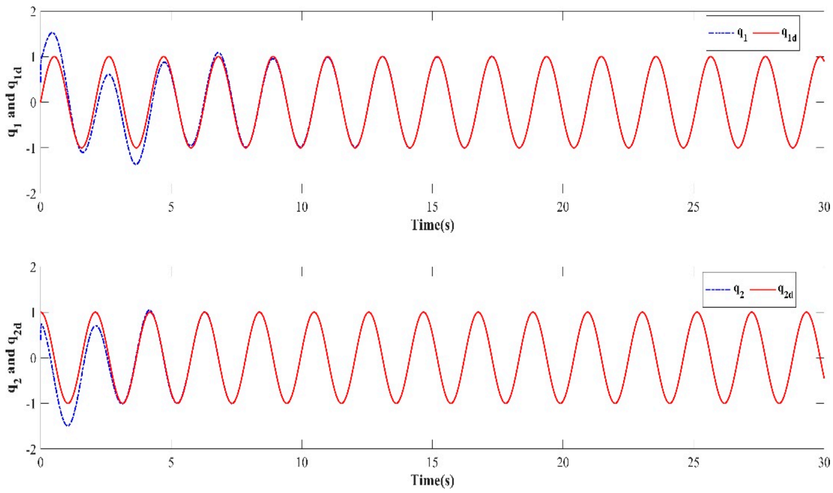

Figure 10. Trajectory tracking effect of the tenth iteration.

It can be seen from Figures 10 and 11 that when an external disturbance was added to the dynamic system of the manipulator, the end position of the manipulator could still achieve high-precision tracking of the desired trajectory after the preset convergence time. By comparing Figure 7 with Figure 10, it can be seen clearly that after the external disturbance was added to the system, the absolute average error value and varying trend after the preset convergence time did not change. This indicates that the addition of an external disturbance to the mechanical arm dynamics system had no effect on the iterative convergence accuracy and convergence speed. The proposed controller was able to achieve high-precision trajectory tracking within the predefined time. It can therefore be concluded that the iterative learning control algorithm proposed in this paper is strongly robust in the face of bounded external disturbances.

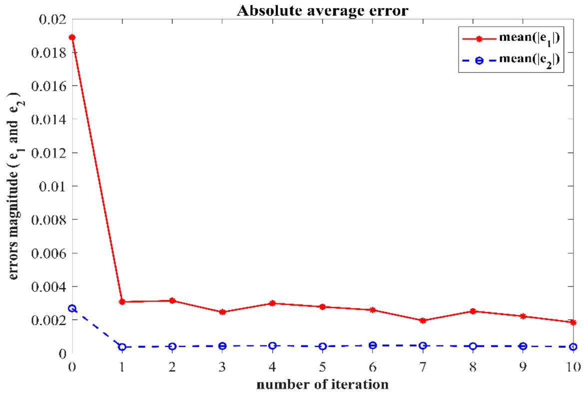

Figure 11. Iterative convergence trend.

6. Conclusions

This paper presented a study into the problem of accurate tracking control of a second-order nonlinear system with arbitrary iterative initial values within a preset time interval. First, a time-varying sliding mode surface with predefined-time convergence and zero initial value characteristics was constructed, and the Lyapunov stability criterion for predefined-time convergence, which is used to evaluate the trajectory tracking error in the sliding mode surface that can converge to the origin at the predefined-time, was given. Second, it was theoretically proven that the predefined convergence time was independent of the initial value of iteration and control parameters. Furthermore, in the process of designing an iterative learning controller, the iterative control problem of trajectory tracking under an arbitrary initial value of iteration was transformed into a time-varying sliding mode surface tracking iterative control problem when the initial iteration value is zero. This establishes a bridge for converting the theory of iterative learning control between the arbitrary initial value and the same initial value. Finally, we presented a design for a PD-type closed-loop iterative learning controller based on a time-varying sliding mode surface. It was proved theoretically that the trajectory tracking error of a second-order nonlinear system can converge and stabilize within the sliding mode surface after learning with a finite number of iterations. This confirmed that the system output was capable of accurately tracking the desired trajectory within a predefined time.

This work expands the application area of iterative learning control theory in practical engineering applications.

Author Contributions: Conceptualization and theoretical proof, C.-W.Y. and S.R.; formal analysis, S.R. funding acquisition, V.B., L.P. and S.M.; methodology, H.Z.; project administration, N.U.; resources, H.Z.; writing—original draft, C.-W.Y.; writing—review and editing, N.U., V.B. and S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Doctoral Grant Competition VSB—Technical University of Ostrava, under Grant CZ.02.2.69/0.0/0.0/19073/0016945; in part by the Operational Programme Research, Development and Education, under Project DGS/TEAM/2020-015; in part by the Partial Discharge Detection in Insulation Systems, National Centre for Energy, under Project TN01000007. This work also received partial support from Taif University Researchers Supporting Project number (TURSP-2020/144), Taif University, Taif, Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors acknowledge the support received from the Doctoral Grant Competition VSB—Technical University of Ostrava, under GrantCZ.02.2.69/0.0/0.0/19073/0016945; in part by the Operational Programme Research, Development and Education, under Project DGS/TEAM/2020-015; in part by the Partial Discharge Detection in Insulation Systems, National Centre for Energy, under Project TN01000007. This work also received partial support from Taif University Researchers Supporting Project number (TURSP-2020/144), Taif University, Taif, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lee, J.H.; Lee, K.S. Iterative learning control applied to batch processes: An overview. *Control Eng. Pract.* **2007**, *15*, 1306–1318. [CrossRef]
2. Li, G.; Lu, T.; Han, Y.; Xu, Z. Adaptive iterative learning control for high-order nonlinear systems with random initial state shifts. *ISA Trans.* **2022**, *130*, 205–215. [CrossRef] [PubMed]
3. Sun, S.-T.; Li, X.-D. Quantized Iterative Learning Control for Nonlinear Switched Discrete-Time Systems with Actuator Saturation. In Proceedings of the 2022 IEEE 11th Data Driven Control and Learning Systems Conference (DDCLS), Chengdu, China, 3–5 August 2022; pp. 1297–1302.
4. Cheng, X.; Jiang, H.; Shen, D.; Yu, X. A Novel Adaptive Gain Strategy for Stochastic Learning Control. *IEEE Trans. Cybern.* **2022**. [CrossRef] [PubMed]
5. Liu, T.; Ding, Y.; Wang, P.; Zhao, K.; Jia, J. Stability Control of Transport Robot Based on Iterative Learning Control. *J. Phys. Conf. Ser.* **2022**, *2173*, 012061. [CrossRef]
6. Liu, J.; Jia, C. Direct and Indirect Technique Routes of Convergence Analysis for Discrete-time Iterative Learning Control. In Proceedings of the 2022 IEEE 11th Data Driven Control and Learning Systems Conference (DDCLS), Chengdu, China, 3–5 August 2022; pp. 899–903.
7. Cheng, X.; Wang, H.; Wang, Q.; Feng, S. Rapid iterative learning algorithm of nonlinear time-delay system with initial deviation. *Int. J. Electr. Eng. Educ.* **2020**, 0020720920940577. [CrossRef]
8. Ma, F.; Li, C. Open-closed-loop PID-type iterative learning control for linear systems with initial state error. *J. Vib. Control* **2011**, *17*, 1791–1797.
9. Riaz, S.; Lin, H.; Waqas, M.; Afzal, F.; Wang, K.; Saeed, N. An Accelerated Error Convergence Design Criterion and Implementation of Lebesgue-p Norm ILC Control Topology for Linear Position Control Systems. *Math. Probl. Eng.* **2021**, *2021*, 5975158. [CrossRef]
10. Liu, F.; Zhang, K. PDα-Type Iterative Learning Control with Initial State Learning for Fractional-Order Systems. *Xibei Gongye Daxue Xuebao/J. Northwestern Polytech. Univ.* **2021**, *39*, 400–406. [CrossRef]
11. Riaz, S.; Lin, H.; Akhter, M.P. Design and implementation of an accelerated error convergence criterion for norm optimal iterative learning controller. *Electronics* **2020**, *9*, 1766. [CrossRef]
12. Yang, J.; Hang, M.; Lin, Y.; Zhang, Q. Adaptive state compensation using parameterized iterative learning control for periodic velocity ripple of permanent magnet linear motor. In Proceedings of the 2009 IEEE International Conference on Industrial Technology, Victoria, Australia, 10–13 February 2009.
13. Riaz, S.; Lin, H.; Elahi, H. A novel fast error convergence approach for an optimal iterative learning controller. *Iintegr. Ferroelectr.* **2020**, *213*, 103–115. [CrossRef]
14. Li, H.; Song, L.; Jiang, X.; Shi, H.; Su, C.; Li, P. Robust Model Predictive Control for Multi-phase Batch Processes with Asynchronous Switching. *Int. J. Control Autom. Syst.* **2022**, *20*, 84–98. [CrossRef]
15. Sun, M.X.; Bi, H.B.; Zhou, G.L.; Wang, H.F. Feedback-aided PD-type Iterative Learning Control: Initial Condition Problem and Rectifying Strategies. *Acta Autom. Sin.* **2015**, *41*, 157–164.
16. Lv, Q. Adaptive iterative learning control for inhibition effect of initial state random error. *Zidonghua Xuebao/Acta Autom. Sin.* **2015**, *41*, 1365–1372.
17. Riaz, S.; Lin, H.; Mahsud, M.; Afzal, D.; Alsinai, A.; Cancan, M. An improved fast error convergence topology for PD α-type fractional-order ILC. *J. Interdisciplinary Math.* **2021**, *24*, 2005–2019. [CrossRef]
18. Yan, Q.Z.; Sun, M.X.; Cai, J.P. Reference-signal Rectifying Method of Iterative Learning Control. *Acta Autom. Sin.* **2017**, *43*, 1470–1477.
19. Chien, C.J.; Hsu, C.T.; Yao, C.Y. Fuzzy system-based adaptive iterative learning control for nonlinear plants with initial state errors. *IEEE Trans. Fuzzy Syst.* **2004**, *12*, 724–732. [CrossRef]
20. Sun, M.; Wu, T.; Chen, L.; Zhang, G. Neural AILC for Error Tracking Against Arbitrary Initial Shifts. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *29*, 2705–2716. [CrossRef] [PubMed]

21. Sun, M.X.; Huang, B.J.; Zhang, X.Z. PD-type iterative learning control for a class of uncertain time-delay systems with arbitrary initial states. *Control Theory Appl.* **1998**, *6*, 853–858.
22. Liu, Y.; Niu, Y. Sliding mode control for uncertain switched systems subject to state and input delays. *Trans. Inst. Meas. Contr.* **2018**, *40*, 3232–3238. [CrossRef]